faster than thought

A SYMPOSIUM ON DIGITAL COMPUTING MACHINES

Edited by B. V. BOWDEN
M.A., Ph.D (Cantab)

With a Foreword by
The Right Hon. The Earl of Halsbury
Managing Director of the National
Research Development Corporation

PITMAN

IN this unique symposium the contributions of twenty-four well-known experts give a clear account of modern digital computing machines, their history, theory and design, and their application to industry, commerce and scientific research.

The Right Hon. the Earl of Halsbury says in his Foreword—"It is undoubtedly the best general account yet written. There is something in it for everyone."

CONTRIBUTORS

Miss M. Audrey Bates, Ferranti, Ltd., Manchester.

Dr. J. M. Bennett, Ferranti, Ltd., Manchester.

Dr. A. D. Booth, Electronic Computation Laboratory, Birkbeck College.

Dr. B. V. Bowden, Ferranti, Ltd., Manchester.

Mr. R. H. A. Carter, Telecommunications Research Establishment.

Mr. E. H. Cooke-Yarborough, Atomic Energy Research Establishment, Harwell.

Mr. A. E. Glennie, Research Establishment, Fort Halstead.

Dr. S. H. Hollingdale, Royal Aircraft Establishment, Farnborough.

Dr. T. Kilburn, Manchester University.

Mr. S. Michaelson, Imperial College of Science and Technology.

Dr. G. Morton, London School of Economics and Political Science.

Mr. B. W. Pollard, Ferranti, Ltd., Manchester.

Miss Cicely M. Popplewell, Royal Society Computing Laboratory, Manchester University.

Dr. D. G. Prinz, Ferranti, Ltd., Manchester.

Dr. R. S. Scorer, Imperial College of Science and Technology.

Mr. J. B. Smith, Ferranti, Ltd., Edinburgh.

Mr. R. Stuart-Williams, R.C.A. Research Laboratories, Princeton, New Jersey, U.S.A.

Mr. B. B. Swann, Ferranti, Ltd., Manchester.

Mr. C. Strachey, National Research Development Corporation.

Dr. K. D. Tocher, Imperial College of Science and Technology.

Dr. A. M. Turing, F.R.S., Assistant Director, Royal Society Computing Laboratory, Manchester University.

Dr. A. M. Uttley, Telecommunications Research Establishment.

Dr. M. V. Wilkes, University Mathematical Laboratory, Cambridge.

Prof. F. C. Williams, O.B.E., F.R.S., Director, Royal Society Computing Laboratory, Manchester University.

42s. net

FASTER THAN THOUGHT

Frontispiece

ADA AUGUSTA
The Countess of Lovelace

FASTER THAN THOUGHT

A SYMPOSIUM ON DIGITAL COMPUTING MACHINES

EDITED BY

B. V. BOWDEN

M.A., Ph.D. (Cantab.)

With a Foreword by

THE RIGHT HON. THE EARL OF HALSBURY

Managing Director of the National Research Development Corporation

LONDON

SIR ISAAC PITMAN & SONS LTD.

First published 1953
Reprinted 1955
Reprinted 1957
Reprinted 1963

SIR ISAAC PITMAN & SONS LTD.
PITMAN HOUSE, PARKER STREET, KINGSWAY, LONDON, W.C.2
THE PITMAN PRESS, BATH
PITMAN HOUSE, BOUVERIE STREET, CARLTON, MELBOURNE
22–25 BECKETT'S BUILDINGS, PRESIDENT STREET, JOHANNESBURG

ASSOCIATED COMPANIES
PITMAN MEDICAL PUBLISHING COMPANY LTD.
46 CHARLOTTE STREET, LONDON, W.1
PITMAN PUBLISHING CORPORATION
20 EAST 46TH STREET, NEW YORK 17, NEW YORK
SIR ISAAC PITMAN & SONS (CANADA) LTD.
(INCORPORATING THE COMMERCIAL TEXT BOOK COMPANY)
PITMAN HOUSE, 381–383 CHURCH STREET, TORONTO

MADE IN GREAT BRITAIN AT THE PITMAN PRESS, BATH
F3—(T.725)

FOREWORD

It is always a privilege to be present at the birth of a new discovery or invention in pure or applied science, a privilege which is at first accessible only to those actually engaged on the work concerned.

Preoccupation with the serious mental striving that is entailed in winning new knowledge leaves little time for writing popular accounts of what is being done, and the student who wishes to follow in the footsteps of the pioneers finds that he must delve through pages of highly technical literature written in a new and unfamiliar jargon *by* those who are doing work *for* those who are doing similar creative work in the same field. Gradually, however, the strain eases, the scaffolding is removed, and it becomes possible for students to see a little of the new building. Didactic exposition, with its own problems of what to teach and how to teach it, then begins.

Some four years ago, finding myself under the necessity of acquiring a certain limited knowledge of how Electronic Digital Computers function, I turned to the patent literature as the only comprehensive digest of information on the subject, one moreover written by draughtsmen whose special skill it is to effect an extreme clarity and precision of description. Notwithstanding the lucky circumstance of having this literature readily available, I found the mental struggle a severe one. Subsequent requests by colleagues to write a "child's guide to computers" left me acutely aware of how difficult the task was, and perusal of various popular expositions written since leaves me equally unsatisfied. I was therefore much interested when Dr. Bowden told me that he was producing a book on the subject himself.

It is undoubtedly the best general account yet written. There is something in it for everyone. For the general reader there is a fascinating account of the intellectual adventures of Charles Babbage and Lady Lovelace a hundred years ago, while for the curious the problems involved in arranging for one of these machines to play chess or draughts are lucidly explained and illustrated by actual games which machines have played against men.

The more serious minded will, however, be able to delve a little deeper into the common symbolism in which arithmetic on the one hand and logic on the other are written, and to learn something of the binary scale in terms of which "machine thinking" proceeds. The physicist with a general knowledge of electronics will be interested in the account of some simple circuit elements which are basic in all machine designs, and the electronic engineer will find a most useful compendium on the detailed design of a number of the leading machines in this country written in each case by the designers themselves or their close associates.

The potential user will find his problems discussed at great length, problems that reside less in the design of the machines themselves than in how to utilize them as part of a commercial organization. Popular accounts, appearing in the Press and elsewhere, have suggested that no problem is beyond the wit of these Electronic Brains; this may be true, but as the author shows, the problem of how to apply them to specific tasks may well surpass the wit of man. These portions of the book display clearly both the opportunities which computers offer and the difficulties which are associated with them and which challenge the business executive who wishes to experiment with them. For good measure the author has thrown in the flow sheet of a P.A.Y.E. calculation.

Finally, for those who relish it, Dr. Bowden's mastery of the ludicrous interpolation is always lurking round the next page as an encouragement to read on.

HALSBURY

PREFACE

The age of chivalry is gone. That of sophisters, economists, and calculators, has succeeded; and the glory of Europe is extinguished for ever—Edmund Burke (November, 1792).

During the last year or two most people must have heard of the remarkable devices often called "Electronic Brains"; every schoolboy knows that there are in existence some very complicated machines which are capable of astounding feats of arithmetic. This book contains descriptions of several of these monsters, an explanation of the way they work, and several essays describing how they can be used. We shall refer to them as "electronic digital computers," a name which describes them more accurately and is less contentious than the one which popular usage has favoured.

The amount of computation which has to be done these days seems to increase rapidly from year to year. The organization of business gets more and more complicated; the welfare state can only be run efficiently on a diet of numbers; it demands them from everyone, and we are fast becoming a nation of clerks. Any machines which can help to do some of the work will be of very great importance. Quite apart from this tendency to complexity in the organization of our society, we find that modern engineering is constantly becoming more complicated; the design of high-speed aircraft, for example, poses all kinds of problems which can only be solved by doing most elaborate calculations. One big aircraft company has doubled the number of computers in its employment every year since 1947. They are now using more than 500 horse power to drive their calculating machines, and this estimate, as they freely admit, makes no allowance at all for friction in their slide rules.

This is the latest and most spectacular development in a very old subject, for arithmetic, which is usually thought to be tedious and uninteresting, has always been of immense importance to science, commerce and engineering. Only a minority of mankind is deeply interested in numerical work as such, but in each generation there seem to have been a few devoted souls who have understood its importance; they have usually failed to win the appreciation of their contemporaries, and since each of them has conceived himself to possess a virtual monopoly of revealed truth, they have sometimes anathematized each other with all the ardour and enthusiasm of theologians, who share many of the same occupational hazards.* All the world knows that Newton's theory of gravitation could never have been produced without the numerical analysis of the orbits of the planets to which Kepler devoted his life, but few appreciate the fact that Kepler in his turn was wholly dependent on the logarithms which had just been produced by Napier and Briggs, and that Kepler's *Ephemeris* was dedicated as of right to "The Illustrious Baron Napier," "To whom," said David Hume, "the title of Great Man is more justly due than to any other whom Scotland ever produced."

"The Illustrious Baron" himself had few illusions about the difficulty and importance of arithmetic. In the introduction to his first book of logarithms he wrote, "Seeing that there is nothing (right well beloved students of the mathematics) that is so troublesome to mathematical practice, nor that doth more molest and hinder calculations, than the multiplications, divisions, square and cubical expansions of great numbers, which beside the tedious expense of time, are for the most part subject to many slippery errors, I began to consider in my mind by what certain and ready art I might remove these hindrances . . ." So far so good; most people would agree with his generalizations and be thankful for the tables which were to help in these operations, but few would react with the fanaticism of Briggs, who, when this book fell into his hands "cherished it as the apple of his eye." "It was ever in his bosom or in his hand, or pressed to his heart, as with eager eyes and mind absorbed, he read it again and again." And this, remember, was a book of logarithms! But log tables, valuable and indispensable

* During the uneasy peace which succeeded our interminable wars with Spain at the beginning of the last century, Captain Smyth, R.N., made a survey of the Mediterranean. He received a courtesy visit from a Spanish Captain, who gave him a silver salver as a souvenir of his visit. King's Regulations, which had been drawn up in detail by Mr. Pepys, made no provision for reciprocation in kind or for charging such gifts against petty cash, so Captain Smyth gave his visitor a handsome leather-bound set of the *Nautical Almanac* (see page 25). This work had been compiled under the direction of that versatile genius, Thomas Young, whose real interest lay in the development of the wave theory of light and the interpretation of the Rosetta stone. It was notoriously unreliable—it omitted 29th February altogether one leap year, and no Englishman ever dared to use it. The Spaniard, who seemed to be unaware of its reputation, sailed away and was never heard of again, but Captain Smyth came safely home, using the Italian and French Navigational tables. "Peace hath her victories." This must rank among the most sophisticated operations ever undertaken by the Royal Navy. The story is to be found in the *History of the Royal Astronomical Society*, but just think what Bunyan would have made of it!

though they have been for three hundred years, will not handle the computations of today, and the demand for machines is becoming more and more insistent.

The principles on which all modern computing machines are based were enunciated more than a hundred years ago by a Cambridge mathematician named Charles Babbage, who devoted his life and fortune to an unsuccessful attempt to construct one. Modern developments in electronics have made his dream come true in the last decade, and there are now a dozen or more machines in the world which do all and more than he expected. These machines amaze us all; they can add up a column of 500 numbers while a mathematician pronounces the word "addition"; they can compute the sines of a dozen angles to twelve decimal places while he mutters the slightly longer word "abracadabra," and they do more arithmetic than a man can do in a day "while one with moderate haste might tell a hundred."

They have already done invaluable work in astronomy, in physics and in engineering, as well as in pure mathematics; they may help to make weather forecasting into an exact science; they bid fair to revolutionize accountancy and book-keeping and the analysis of Government statistics, thereby enabling economists to understand in detail the course of world economic trends. Time alone will show if this will help a government to forecast cycles of boom and slump and perhaps to control them; alternatively perhaps the figures which machines produce will encourage politicians and planners to befuddle themselves more methodically and more completely than ever before. Machines are already learning to play chess and draughts; before long there may be transatlantic matches played entirely automatically by wireless. Professor von Neumann has proved that in theory these machines can reproduce themselves, but no one has yet explained this fact to a machine, and so far they have never shown any signs of taking the initiative.

A rough count showed that about 150 digital computers are being built at this moment, most of them in universities and other research establishments. It will be interesting to see if these machines play in the next decade the part of the cyclotrons and high voltage generators in the "thirties." In those days every university had to have a cyclotron on the campus; they were mysterious and expensive and they gave tone to the place; they impressed distinguished visitors and attracted endowments; their construction gave the whole of the physics department plenty of healthy exercise, and kept them happy, out of mischief and covered in oil; the united efforts of the staff were required to keep the machine on the verge of operation, and those who so wished could postpone into the indefinite future the embarrassing decision as to what was to be done with the machines when they actually started to work. Many of the physicists who made their mark in the war learnt their trade by building cyclotrons, but these machines have become so big and so expensive nowadays that they can only be made by large firms of professional engineers. There is much to be said for digital computers as research projects for the time being; they are not so expensive as cyclotrons, they are much less messy, they are even more incomprehensible, and perhaps before very long they too will have to be taken over by the big firms.

It seems probable that we shall have a second Industrial Revolution on our hands before long. The first one replaced men's muscles by machines, and every worker in England now has an average of more than 3 horse power to help him. In the next revolution machines may replace men's brains and relieve them of much of the drudgery and boredom which is now the lot of so many white collar workers. No one has yet proposed a unit in which to measure "brain power," so one cannot express in numerical terms the help which the next generation of clerks may expect to receive from machines.

It has already become clear that it will be very difficult to train the operators who will be needed to look after the machines of the new era, but complicated though they are and always will be, it would be a mistake to assume that each digital computer is a kind of Moloch, which has to be fed on mathematicians. For one thing they will keep tomorrow's arithmeticians happy. Nowadays many of these dedicated men spend their time in computing prime numbers. The search for the largest known prime is a hobby which is at least as useful and interesting as playing bridge, and computing machines have helped enormously. The reader will not be surprised to hear that nowadays the biggest primes are found in America. The largest which has been discovered so far (January, 1953) consists of 2281 consecutive "ones," when it is expressed in the binary scale (see page 33).

Ferranti Ltd., who had built a large digital computer for the University of Manchester, made a simple demonstration computer called "Nimrod" for the Science Exhibition at the Festival of Britain. It was very limited in its abilities; all it could do was to play a rather elementary game, but it attracted a great deal of interest, and many of the people who saw it asked for more information than the demonstrators could give. This book originated in an attempt to answer some of these questions and it is, in retrospect, a startling commentary on my inexperience as an author that I originally hoped to finish it in time for the end of the Festival. The book has been growing in size and scope ever since I began it. I proposed to describe the present state of the art in simple terms and to make a tentative forecast of probable progress, but I soon found that it would be impossible for me to cover the whole subject myself, and I have been fortunate enough to persuade several people whose experience in the field is far greater than mine to contribute to the book, which contains, I believe, an account written by the designer of every machine which was being built in England in 1951 (with two exceptions), and essays by several economists, statisticians, engineers, physicists and mathematicians.

I decided to concentrate on work which is now being done in this country; it was more convenient to do so; far more work has been done over here than most people realize, and several excellent books describe contemporary developments in America.

The early history of these machines and the story of poor Babbage's struggles is very interesting. We owe our best account of Babbage's "Engines" to the Countess of Lovelace, who was a mathematician of great competence and one of the very few people who understood what Babbage was trying to do. Her ideas are so modern that they have become of great topical interest once again, and since her paper has long been out of print (it appeared more than a hundred years ago) it has been reproduced as an appendix to this book. Lady Lovelace's grand-daughter, the Right Hon. Lady Wentworth, has very kindly allowed me to read many of Lady Lovelace's most interesting papers; I was so surprised by the connexion that I found between digital computers and thoroughbred horses that I have given a brief account of the story, for further details of which the reader is referred to Lady Wentworth's own books.

After I had finished the book, I saw a microfilm of a life of Babbage which had been written by his executor, the late Mr. L. H. D. Buxton. Mr. Whitwell of the Powers Samas Company found the manuscript in the Museum of the History of Science in Oxford. It contains a more detailed account of the construction of Babbage's Engines than any I have seen elsewhere, and it is to be hoped that the material will some day be published.

The list of Babbage's inventions is endless;* but in spite of one's sympathy for him as a misunderstood genius, one cannot help feeling that he brought many of his troubles on himself. A man who, when he was trying to enlist the support of the Royal Society for his engines, could publicly describe its Council as "a collection of men who elect each other to office, and then dine together at the expense of the Society to praise each other over wine, and to give each other medals" would probably find life difficult in any age. His friends often saw with despair that he had decided yet again to scrap everything he had done so far and start all over again. Nevertheless, if he had done nothing more, the improvements which he effected in the art of machining and in machine shop practice more than justified the £17,000 which the British Government gave him to build his Engines.

It is always difficult when one is trying to explain a very technical subject to know how much detail to give; almost every chapter in the book might have been expanded into a large book on its own. Many people have learnt enough about pulse circuits and electronics in general to appreciate and follow a straightforward attempt to take them behind the scenes and give them some idea of the circuits which are used in these machines. Chapters 2 and 3 are intended for them, but they do not cover the subject at all comprehensively; those who have little or no experience of electronics are advised to miss out these chapters altogether if they find them hard to understand. If they will accept the fact that electronic adders, multipliers and what-not can be built, that is all they really need in order to follow the rest of the story.

A certain amount of repetition is inevitable in a series of essays such as are to be found in this book, and occasional differences of opinion between authors are possible too. It seemed to me that the book would lose more than it would gain if the contributions were edited to remove them.

I have borrowed freely from the writings of several authors, and I have tried to acknowledge the source when I could remember it. Dr. Johnson thought that a man could turn over a library to make a book; even in these degenerate days it is accounted plagiarism to copy from one author. Academic tradition now accepts the con-

* He once tried to investigate statistically the credibility of the biblical miracles. In the course of his analysis he made the assumption that the chance of a man rising from the dead is 1 in 10^{12}. This must rank with Sir Arthur Eddington's "perfectly smooth elephant, whose weight may be neglected" among the more remarkable postulates of Cambridge mathematicians.

vention that to crib from more than two counts as research, and if, as I have done, one can enlist the help of several collaborators, the project acquires status and becomes quite respectable. I have tried not to follow the precedent established by the author of a famous treatise on Chinese Metaphysics, who, as the reader will doubtless recall, read the articles in the encyclopedia on China and on Metaphysics, and "combined the two." Much of this book derives from the work of those prolific authors "Anon" and "Ibid" who have done so much to put our English platitudes on a sound literary basis.

I must express my thanks to all my collaborators; to Lord Halsbury for writing the foreword; to Lady Wentworth who gave me so much information about Lady Lovelace, and who allowed me to reproduce the portrait which has been used as a frontispiece; I am also indebted to Miss Draper who read all the Lovelace papers and gave me a great deal of help. I must thank Miss Dyke for preparing the flow sheets which I used in Chapter 22. Dr. Gilles and Mr. Whitwell told me the story of Dr. Comrie; Dr. Bullard found some of Babbage's writing in the archives of the National Physical Laboratory; and Professor Aitken, Mr. W. Klein, Dr. van Wijngaarten, Dr. Stokvis, Mr. Seeber, Mr. Ferris and Dr. Gabor gave me much of the information on which Chapter 26 is based. The Portrait of Babbage is included by courtesy of the Director of the Science Museum, South Kensington. I have discussed the whole book with my colleagues here in Moston. In particular I must acknowledge the help of Miss Bates and Mr. Leech, who helped with the editorial work and with proof reading. My secretaries have retyped the manuscript so often that they must almost know it by heart.

The frequent references in the text to the Manchester machine are due primarily to the fact that I could explain a principle only by describing its impact on one particular machine. For purposes of illustration I have used the machine with which I am most familiar, but many other machines embody the same ideas.

Finally I must acknowledge the co-operation of the Publishers, who have accepted with apparent equanimity the fact that progress in the subject is so rapid that corrections are necessary even while the type is being set. It is impossible to produce a book on this subject which is not out of date in some respects before it appears.

B. V. Bowden

Manchester
January, 1953

LIST OF CONTRIBUTORS

Miss M. Audrey Bates, Ferranti Ltd., Moston, Manchester (*Chapter* 25)

Dr. J. M. Bennett, Ferranti Ltd., Moston, Manchester (*Chapters* 5, 17, 20)

Dr. A. D. Booth, Director of the Electronic Computation Laboratory, Birkbeck College, London (*Chapter* 13)

Dr. B. V. Bowden, Ferranti Ltd., Moston, Manchester (*Chapters* 1–4, 14, 22, 25, 26

Mr. R. H. A. Carter, Telecommunications Research Establishment, Malvern (M.O.S.) (*Chapter* 10)

Mr. E. H. Cooke-Yarborough, Atomic Energy Research Establishment, Harwell (M.O.S.) (*Chapter* 9)

Mr. A. E. Glennie, Research Establishment, Fort Halstead (M.O.S.) (*Chapters* 5, 19)

Dr. S. H. Hollingdale, Head of the Mathematical Services Department, Royal Aircraft Establishment, Farnborough (M.O.S.) (*Chapter* 12)

Dr. T. Kilburn, Senior Lecturer, Electrical Engineering Dept., Manchester University (*Chapter* 6)

Mr. S. Michaelson, Imperial College of Science and Technology, London (*Chapter* 11)

Dr. G. Morton, Lecturer in Economics, London School of Economics and Political Science (*Chapter* 23)

Mr. B. W. Pollard, Ferranti Ltd., Moston, Manchester (*Chapter* 2)

Miss Cicely M. Popplewell, Staff Member of the Royal Society Computing Laboratory, Manchester University (*Chapter* 24)

Dr. D. G. Prinz, Ferranti Ltd., Moston, Manchester (*Chapter* 15)

Dr. R. S. Scorer, Lecturer, Department of Meteorology, Imperial College of Science and Technology, London (*Chapter* 18)

Mr. J. B. Smith, Ferranti Ltd., Crewe Toll, Edinburgh (*Chapter* 15)

Mr. R. Stuart-Williams, Sometime of Ferranti Ltd., Moston, Manchester, now at the R.C.A. Research Laboratories, Princeton, New Jersey, U.S.A. (*Chapter* 16)

Mr. B. B. Swann, Ferranti Ltd., Moston, Manchester (*Chapter* 21)

Mr. C. Strachey, National Research Development Corporation (*Chapter* 25)

Dr. K. D. Tocher, Imperial College of Science and Technology, London (*Chapter* 11)

Dr. A. M. Turing, F.R.S., Assistant Director of the Royal Society Computing Laboratory, Manchester University (*Chapter* 25)

Dr. A. M. Uttley, Telecommunications Research Establishment, Malvern (M.O.S.) (*Chapter* 10)

Dr. M. V. Wilkes, Director of the University Mathematical Laboratory, Cambridge (*Chapter* 7)

Professor F. C. Williams, O.B.E., F.R.S., Professor of Electrical Engineering and Director of the Royal Society Computing Laboratory, Manchester University (*Chapter* 6)

Chapter 8 is reprinted from *Engineering* by kind permission of the Publishers.

CONTENTS

2—(T.725)

PART THREE

APPLICATIONS OF ELECTRONIC COMPUTING MACHINES

INSETS

LIST OF PLATES

PART ONE

THE HISTORY AND THEORY OF COMPUTING MACHINES

Chapter 1

A BRIEF HISTORY OF COMPUTATION

God made the Integers, all the rest is the work of man—
LEOPOLD KRONECKER

THIS IS NOT THE PLACE to attempt an account of the development of arithmetic. The art of computation was studied thousands of years ago, and some of the calculations of the ancients, for example those with which they predicted eclipses of the sun, command our admiration today. Numerical work has always been the basis of commerce. The oldest written "documents" which survive in the world were produced about 5,000 years ago at Erech in Mesopotamia by the Sumerians on clay tablets; they were primitive books of accounts, and they show the severely practical reasons which prompted the invention of the Sumerian syllabary.* Thousands of years later, when movable type was invented and the first printing press was set up in Europe by Gutenberg, most of the books which he published at first were text-books of commercial arithmetic.

Nevertheless a few hundred years ago the art of computation was neither commonly understood nor widely practised. In 1662, Pepys, who was then in charge of the contracts branch of the Admiralty, found it necessary to put himself to school, and to rise by candlelight at four o'clock in the morning in order to learn his multiplication tables. He had been to Cambridge; and was, by the standards of his time a well-educated man, in later life he became President of the Royal Society and a friend of Newton, but when he was Clerk of the Acts he found himself quite unable to understand the simple computations which had to be done when buying timber for the King. In those days schoolboys seldom went beyond "two times two." Since then computation has become more widely understood as the need for it has grown, but, as Barlow(1) remarked a hundred years ago, "Calculation is laborious and unremunerative" and, he added rather pathetically, "a moderate skill in computation

* One of the earliest written documents which can be described as "non-quantitative" contains a lament for the "good old days," and a statement that the world is going to the dogs because children no longer obey their parents.

and a persevering industry are not precisely the qualities a mathematician is most anxious to be thought to possess."

The introduction of logarithms by Napier and Briggs revolutionized ordinary computing; our civilization, dependent as it is on navigation, surveying, and astronomy, could probably not have developed as it did without them. Briggs devoted his life to computing the logarithms which Napier had invented. Both men realized the importance of their work; when they were introduced in Edinburgh, a friend related that they gazed at one another in speechless admiration for a quarter of an hour before the silence was broken.

Few people nowadays appreciate the prodigious amount of labour which went into the production of the first tables of logarithms and of trigonometrical functions such as sines and tangents. Many forgotten arithmeticians devoted their lives to the work; sometimes shamelessly pirating each other's results, a fact which was made all too clear when they reproduced each other's mistakes. For example, thirty individual errors first made in a table published in Europe in 1603, were found in a set of tables printed in Chinese two hundred years later.* The difficulty of reducing the results of an ordnance survey (which is hard and laborious enough under the best of circumstances), if the mathematical tables may be wrong, can hardly be imagined, and there is a story of a ship being wrecked at the Lizard because the Master used a wrong value in a table when working out his position. The calculations were found in his cabin after his death. Subsequent editions of the tables were corrected, but this is a desperately extravagant way of checking the accuracy of logarithmic and trigonometrical tables. It was all very well for de Joncourt to write (as he did in 1762) "Lo, the enraptured arithmetician. Easily satisfied, he asks for no Brussels lace, or a coach and six. To calculate contents his liveliest desires, and obedient numbers are within his reach." Few of his contemporaries shared his enthusiasm; most of them were only too well aware of the labour involved in the computations which had to be done if science and mathematics were to progress and if commerce was to continue, and they realized very clearly how much mistakes in a bill of lading or in a table of logarithms could cost.

* In his introduction to the new edition of Chambers *Six Figure Tables*, Dr. Comrie remarked that he had left three deliberate mistakes, each of one digit in the last place "as a trap for would-be plagiarists." Is this really fair?

For centuries inventors have devoted themselves to the development of calculating machines which would reduce the drudgery of computation and improve the reliability of the results.

Two distinct types of machine have been evolved. The first is typified by a slide rule, a planimeter or a differential analyser. These machines represent the magnitude of numbers by some such physical quantity as a length or perhaps a voltage. They are known as analogue machines, and are very useful, particularly in calculations where extreme accuracy is not required. An ordinary slide rule, for instance, easily yields results correct to one per cent; a good twenty-inch rule, carefully used, can be relied upon to about one part in 1,000; large slide rules have been made in the form of spirals on cylinders which are good to perhaps one part in 10,000, but no one has ever made a slide rule which would give results more accurately than one part in 100,000. The same order of accuracy can be obtained from other analogue machines. On the other hand the accuracy which an arithmetic process can yield is limited only by the number of figures (or digits) which one cares to use. We shall not discuss analogue machines in this book, but we shall confine our attention entirely to those machines which work in "digits" just as one does on paper, and are therefore called *digital* machines.

The abacus, or counting frame was invented many thousands of years ago, and, in fact the word *calculate* is derived from the pebbles or *calculi* on the wires of an abacus. This instrument is said to have been introduced into Europe about a thousand years ago by Gerbert, who afterwards became Pope Sylvester II. He is supposed to have learned of it from the Arabs in Spain. Until his time Europeans had been concerned to investigate the properties of individual numbers rather than their combinations. The numbers 3 and 6 for example were thought to contain all the secrets of nature, and Peter Abelard called the science of arithmetic *nefarium*. Sylvester had an immense popular reputation as a magician, and he was reputed to have made a bronze head which would answer *yes* or *no* to any question propounded to it. Many legends based upon his life were used as a warning to Christians to avoid the Black Arts. The difficulty of doing arithmetic by using Roman numerals is such that the first systems of numbers were intended to record the contents of the wires on an abacus. It is still quite hard to multiply MDCXVIII by MDCCCCLII directly. With the introduction of Arab numerals into Europe the abacus has come to be regarded as a toy, but it is still used in the far East for everyday computations.

In 1642 the first simple digital calculating machine was built by Blaise Pascal, who was then only 19 years of age. He hoped that it would be of assistance to his father, who was a customs officer in Rouen. Soon afterwards Leibnitz designed the so-called *stepped wheel* which is still to be found in a few machines, for, as he wrote (in 1671): "It is unworthy of excellent men to lose hours like slaves in the labour of calculation which could safely be relegated to anyone else if machines were used." Leibnitz's machine was completed in 1673, and it was exhibited before the Royal Society of London; it attracted widespread attention, but unfortunately its operation was never dependable.

In 1878 the Swedish engineer Odhner invented his pin-wheel method of adding numbers from one to nine. His patents were taken up in Germany and incorporated in the best known of all hand calculating machines, the *Brunsviga*, which is used in large numbers in banks, in business offices and in mathematical laboratories. The speed with which an operator can perform the normal processes of arithmetic with one of these machines is far greater than he could ever achieve with pen and ink.

However, it is possible to exaggerate the speed which can be achieved in computing by using a desk calculator. In 1946 the American Army staged a competition involving ordinary arithmetic operations between one of their Japanese clerks, Kihoshi Masturaki, who used an abacus, and Private Wood, who used modern desk calculating machinery. The Japanese won every time. It is, therefore, not surprising that there should be in Japan, a flourishing society for popularizing the use of the abacus. Its president is chairman of the well-known *Hemmi* slide-rule company, and a graduate of Nottingham University.

Desk calculating machines, of the type with which most people are familiar, all require the intervention of the human operator at every step. It is well known that modern machine-tools gain very much in speed because they can perform a sequence of operations entirely automatically once they have been set up. It is reasonable to inquire, therefore, whether it is possible to devise a machine which will do for mathematical computation what the automatic lathe has done for engineering.

The first suggestion that such a machine could be made came more than a hundred years ago from the mathematician Charles

Babbage. Babbage's ideas have only been properly appreciated in the last ten years, but we now realize that he understood clearly all the fundamental principles which are embodied in modern digital computers. His contemporaries marvelled at him though few of them understood him; and it is only in the course of the last few years that his dreams have come true. The rest of this book is devoted to an account of the construction and use of the machines which his vision inspired, but let us interrupt the thread of our story for a moment and give a brief account of Babbage, the man.

He was born in Devonshire in 1792, the son of a banker from whom he inherited a considerable private income. He received a desultory education, but he taught himself mathematics so well that when he went to Cambridge he found to his dismay that he knew more algebra than his tutor. At that time mathematics in Cambridge was still dominated by the influence of Newton, and was quite unaffected by contemporary developments on the Continent, so Babbage joined forces with two of his friends, John Herschel (son of the discoverer of the planet Uranus and later a notable astronomer in his own right) and George Peacock, who afterwards became Dean of Ely. The three of them decided to try to "do their best to leave the world wiser than they found it" and founded the Analytical Society, which became a very influential body, and gave the first impulse to a revival of the study of mathematics in this country after half a century of neglect. In particular, the society objected to the notation devised by Newton for writing the differential coefficient of X as $\dot{X}$ and, as Babbage put it: "They advocated the principles of pure D-ism as opposed to the Dot-age of the University."

Babbage decided that Herschel was certain to be Senior Wrangler, and as he did not care to be second, he refused to take the Tripos at all. Perhaps in this he was a little too impetuous.

Thirty years later another undergraduate of Peterhouse, named William Thomson, better known to fame as Lord Kelvin, was so certain he would be top of the list that he sent his servant to find out who was second. (It was William Thomson.) In spite of the fact that he had no honours degree, Babbage was elected to the Lucasian Chair of Mathematics (Newton's chair) in 1828, and he established another precedent by holding the office for eleven years without giving a single lecture in the University. For the rest of his life his scientific activity was untiring and conspicuous. He was outstanding among his contemporaries because of his insistence on the practical application of science and of mathematics; for example, he wrote widely on the economic advantages of mass production, and on the development of machine-tools.

In 1812 he was sitting in his rooms in the Analytical Society looking at a table of logarithms, which he knew to be full of mistakes, when the idea occurred to him of computing all tabular functions by machinery. The French Government had produced several tables by a new method. Three or four of their mathematicians decided how to compute the tables, half a dozen more broke down the operations into simple stages, and the work itself, which was restricted to addition and subtraction, was done by about eighty computers who knew only these two arithmetical processes. Here, for the first time, mass production was applied to arithmetic, and Babbage was seized by the idea that the labours of the unskilled computers could be taken over completely by machinery which would be quicker and far more reliable. The principles which he proposed to use are very simple and can be understood by reference to the table below. The first column contains the sum of the squares and the cubes of the natural numbers, the second

TABLE I

A TABLE OF DIFFERENCES

N	*D*1	*D*2	*D*3
0			
	2		
2		8	
	10		6
12		14	
	24		6
36		20	
	44		6
80		26	
	70		6
150		32	
	102		6
252		38	
	140		6
392		44	
	184		6
576		50	
	234		6
810		56	
	290		
1100			

See page 142 for a table of differences computed by the Harwell machine and page 346 for a table of differences computed by Menabrea in 1842.

column the differences between the successive members of the first column, the third the differences between successive differences, and so on.

The numbers in column $D3$ are equal, so it is clear that additional numbers of column $D2$ can be found by simple addition. They will be 62, 68, 74, 80, etc., so that from them the next numbers of the second column, and ultimately of the first column, can be found. The table of $N = n^2 + n^3$ can therefore be computed by using no process beyond simple addition. Most functions can be expressed as a power series and this method can be used to evaluate them if enough differences are taken.

The same method can also be used to detect errors in tables. Suppose for example in our table we had written 151 instead of 150; then our last column, instead of being full of sixes, would read: 6, 7, 3, 9, 5, 6, 6, a sequence which shows clearly that a mistake has been made.

Babbage proposed to use his *Engine* both for computing new tables, and for checking those which had already been published. He constructed a small working model which he demonstrated in 1822, and this was received with such enthusiasm that he proposed to make a much larger machine, working to twenty decimal places and sixth-order differences, with which, he said, useful tables could be computed.

The Royal Society supported the project and Babbage had an interview with the Chancellor of the Exchequer, at which, as he afterwards maintained, he was promised a subsidy to help to pay for the cost of development. The Government built him a workshop and a special fireproof vault to house the drawings which were prepared, but Babbage took much longer than he had anticipated to complete his designs, and in 1842 official support for the project was withdrawn. Only one member of parliament voted in favour of the continuation of the work to which Babbage had devoted so many years of his life, and upon which he had spent a large part of his fortune. The most strenuous representations by the Royal Society had no effect upon the Government, and construction of the Difference Engine was abandoned.

It was left in Somerset House until it was demonstrated by Babbage at the great International Exhibition of 1862. The machine was displayed in a small dark inconvenient cubby-hole entirely surrounded by carpets and oil cloths, for, as Babbage remarked, "the Commissioners were better qualified to judge of furniture for the feet than of furniture for the head." The machine excited great interest and visitors came from all parts of the world to see it, but poor Babbage had to give almost all the demonstrations himself, and many people were unable to get near. After this Babbage had more trouble with the Government, but a model of the difference engine, which had been built in 1859 for the Registrar General was used in 1863 for calculating the life tables from which insurance companies computed their premiums for many years. Parts of these machines are to be found in several museums, most notably the Science Museum in South Kensington (see Plate II). The idea of a difference engine then seems to have been forgotten for more than half a century.

The most difficult problem which Babbage had to face in making this machine was the "carry over" in addition, and he realized that if a carry was, as it were, "propagated" from one end of a twenty-digit number to the other, changing a whole series of nines to noughts, the machine would be relatively slow. He devoted many years to a study of the problem until he finally produced a mechanical method of simultaneous operation of all "carries," the modern version of which is in use today in all "parallel" computers* (see page 34).

In 1833, while the construction of the machine was suspended for a year, Babbage conceived the idea of his Analytical Engine, which he realized would be far more versatile than the difference engine. It would be able to perform any calculation whatsoever. It was, in fact, the first universal digital computer, as the expression is understood today, and Babbage devoted the rest of his life to an attempt at perfecting it. He had to finance the whole of the work himself, and he was only able to finish part of the machine, though he prepared thousands of detailed drawings from which it could have been made.† His son, Major-General H. P. Babbage,[3] continued the development after his death and built part of the arithmetic unit, which printed out its results directly on paper. Another model was built in Sweden which, as a contemporary account informs us, "could be turned with about the force requisite to operate a small barrel organ."

Babbage was full of most ingenious ideas; for example he devised

* See Chapters 9 and 10.

† In a moment of despair he contemplated abandoning the work, but his mother encouraged him to continue: "Even if it should oblige you to live on bread and cheese." At this time, when beef cost 5d. per pound, Babbage was paying his chief draughtsman a guinea a day.

the method which during the last war became known as *Operational Research*, and applied it to an analysis of the pin-making industry. A similar analysis of the printing trade led to results which so offended his publishers that they refused to accept his books. He said, "Political economists have been reproached with too small a use of facts, and too large an employment of theory . . . let it not be feared that erroneous deductions may be made from recorded facts: the errors which arise from the absence of facts are far more numerous and more durable than those which result from unsound reasoning respecting true data." This last sentence might be taken as the motto of operational-research workers the world over. One of the most remarkable applications which he made of the method was to an analysis of the economics of the Post Office. He showed conclusively that the cost of collecting, "stamping" and delivering a letter was far greater than the cost of transporting it. He therefore suggested that operations of the Post Office should be simplified by the introduction of a flat rate of charges, which should be independent of the distance for which the letter had to be carried. It was as a result of these arguments of his that Sir Rowland Hill was encouraged to introduce the penny post a few years later. He studied the records of the Equitable Life Insurance Company, and published in 1824 the first comprehensive treatise on actuarial theory, and the first reliable "life tables." They were used both in England and in Germany for half a century as the basis of the new and rapidly growing life-insurance business, but were superseded in England about 1870 by a set of tables which had been computed by the Board of Trade on the specially built difference engine which we have already mentioned. Babbage also computed a celebrated table of logarithms, and he bitterly criticized the inaccuracies in published astronomical tables, which, so he said, should have been computed automatically on one of his engines, but, as we shall see, the Astronomer Royal did not agree. Babbage had a life-long interest in inventing and in solving codes and ciphers of all kinds; he made skeleton keys for "unpickable" locks; he devised the method, which is now familiar to everyone, of identifying lighthouses by occulting their lights in a rhythmical manner, and had the mortification of seeing the scheme used for the first time during the Crimean War—by the Russians; he proposed a method of recognizing cycles of wet and dry weather in the annular rings of trees. Fifty years ago this method was rediscovered and it has been used, particularly by Professor Douglas in the University of Arizona, to date the timbers which survive in prehistoric dwellings two or three thousand years old.

Babbage had an amazingly prescient understanding of the uses of his machines. Among the more important large-scale computations which have been done on modern machines are those of the Fourier series which are needed to infer the structure of molecules from their X-ray diffraction patterns (see Chapter 17). It is astonishing to find that he wrote in 1838: "The whole of chemistry, and with it crystallography, would become a branch of mathematical analysis, which, like astronomy, taking its constants from observation, would enable us to predict the character of any new compound, and possibly indicate the source from which its formation might be anticipated."

He set himself up in private practice as a consulting engineer, and became very interested in the development of the railways. He was a friend of Sir Isambard Brunel, chief engineer of the Great Western Railway, and helped him by inventing the dynamometer car, with which he could automatically measure and record the tractive force of the locomotive and the irregularities of the track. He used to run his special train on Sundays, as there were fewer other trains to compete with, but nevertheless the signalling system was so bad that he often found himself heading for another train coming directly towards him on the same track. On more than one occasion he owed his life to the remarkable acuteness of his hearing, which enabled him to get on to a siding and avoid a head-on collision. He suggested the use of the "cow catcher," and he devised the first speedometer; he thought that there should be one in the cab of every locomotive. On one occasion he found himself on Hanwell viaduct on a flat car with no engine, and by holding up a piece of cloth which he had with him he was able to *sail* across the viaduct. He remarked that he thought that he was the first man ever to do this, and it is likely that here again he established a record which will stand for many years to come.

The art of working metals to close tolerances had not reached a state in which the cogwheels and levers he needed could be produced in quantity, so Babbage spent years in improving lathes and gear-cutting tools. He found that it was necessary to use a diamond tool in some of his precision turning, and he seems to have had the idea, so fundamental to modern mass production of "universally interchangeable parts." Had anyone been able to appreciate and to exploit some of his proposals in this field, contemporary machine-shop

PLATE I. CHARLES BABBAGE
From a portrait in the Science Museum, South Kensington
(Crown Copyright Reserved)

practice might have anticipated the developments of the next half century. As it was, his influence was felt only indirectly through the foreman of his workshop who, years later as Sir Joseph Whitworth, became famous as one of the best precision engineers in the world and was responsible for introducing the first series of standard threads.

It gives us some measure of Babbage's achievement if we recall that in 1815 he saw Napoleon Bonaparte on board the *Bellerophon* in Plymouth harbour. This seventy-four gun battleship contained no machinery at all; in fact the only mechanical contrivances on board were the fire pump and the "yard arm bilge pump," both of which were manually operated, and a couple of capstans each of which had to be operated by a hundred and fifty of the crew. This was the best that contemporary engineering could do, yet Babbage had already conceived the principles of the difference engine, which was to revolutionize the computation of mathematical tables throughout the world when it was rediscovered a hundred years later.

Babbage was misunderstood by his contemporaries, and he quarrelled with many of them. For example he tried to reform the Royal Society, "to rescue it," as he put it, "from contempt in our own country, and ridicule in others." One would have thought that any man who embarked on a crusade of this type with such a war cry would have anticipated a certain amount of opposition from someone, but Babbage never seems to have allowed other people's feelings to stand in his way when he had a "cause" to pursue. In the end he seems to have become estranged from almost all his friends; he must have grown old alone.

Babbage blamed the Royal Society for the conditions at the Royal Observatory, Greenwich. He applied on one occasion for a copy of some of the Greenwich observations; when his request was refused he looked further into the matter, and found that there were, in one shop in Thames street, more than five tons of the *Greenwich Tables*, which had been sold at fourpence a pound for making pasteboard. Since they had been printed on good quality paper with wide margins, he was assured by the dealer that they "made capital Bristol Board." Apparently the Astronomer Royal had all unsold copies of the *Tables* as a perquisite of his office. Babbage remarked acidly that no one was better fitted than the Astronomer Royal to decide what should be done with his observations, but he doubted if it were possible to devise a more extravagant method of remunerating a public official than to set up an Observatory and a computing centre and to produce and print astronomical tables merely as a source of wastepaper. It is impossible not to sympathize with him, but it was Airy, the Astronomer Royal, who recommended to the Government that it should refuse further support for the difference engine, and who himself refused to consider the mechanization of the computation of his tables. Airy must have been in good form at the time, since it was he who, in his official capacity as Chief Scientific Adviser to the Queen, warned the Government that the Crystal Palace would fall down when the Royal Salute was fired to welcome the Queen at the opening of the Great Exhibition. Fortunately other experts contradicted the Astronomer Royal on this occasion, but poor Babbage had no remedy, nor had Adams who, after he had predicted the position of the new planet Neptune, was baffled by Airy's recommendation that the search be conducted "in a leisurely and dignified manner," with the result that the planet was first discovered by a less inhibited observer, who had Leverrier's calculations to guide him.

Perhaps Airy was preoccupied with his other responsibilities. As recently as 1950 people still wrote to the Astronomer Royal to have their horoscopes cast and their fortunes told. One of the most amusing stories is told of Flamsteed, the first Astronomer Royal. An old lady who had lost something pestered him to tell her where it was. This so annoyed him that he resolved to teach her a lesson, so he drew a few squares and circles on the ground and told her to dig where he stuck his walking stick. He had, of course, intended to show her the absurdity of the whole thing, but when she dug in the place he suggested, she found everything she had lost, and became a firm believer in astrology for the rest of her life. Flamsteed attributed his success to the direct intervention of the Devil, and it is amusing in retrospect to decide whether the lady was more superstitious than the astronomer.

"In England," Babbage wrote, "those who have hitherto pursued science have in general no very reasonable grounds for complaint; for they should have realized that there was no demand for it, and that it led to little honour, and less profit." Consider the frustration which led him to write: "Occasionally a few simply honest men are to be found upon a committee, they are useful as adjuncts to give a high moral tone to the cause, but the rest of the committee usually think them bores, and when they differ from the worldly members, it is usually whispered that they are crotchety fellows," or: "An abject worship of Princes and an unaccountable appetite for knighthood are probably unavoidable results of placing second-rate men in prominent positions." He once anticipated the experiments of Professor Haldane by spending ten minutes in an oven at a temperature of 265 degrees Fahrenheit—he probably found it a pleasant relief after the controversies of the day.

Babbage was intensely annoyed by the cries of street musicians, who, so he said, made it impossible for him to concentrate on his work. Instead of following the example of a fellow sufferer—Thomas Carlyle—who retreated to a sound-proof room, Babbage embarked on a life-long vendetta against them, and tried to have them prosecuted. This public-spirited action so enraged his contemporaries that jeering children followed him through the streets; drum and fife bands came miles out of their way to serenade him, and indignant citizens who had an hour or two to spare made a point of having a drink at some local hostelry, and then blowing bugles and other instruments under his windows at all hours of the day and night. Fortunately none of them understood his machines well enough to foresee their ultimate impact on society. Mathematicians must be either more understanding or less excitable than weavers, in any event the Luddites left him alone.

It was the tragedy of the man that, though his imagination and vision were unbounded, his judgment by no means matched them, and his impatience made him intolerant of those who failed to sympathize with all his projects. For example he first made a simple model of the difference engine which worked to six decimal places and second-order differences, but as soon as he had shown in 1822 that such a machine was possible, he started to construct a much larger and more complicated engine which was to work to twenty decimal places and sixth-order differences. Had he made a reliable machine which worked to ten places of decimals and second- or third-order differences it would in the hands of a competent computer have revolutionized contemporary table making. The first difference engine which was extensively used in this country (by T. C. Hudson in 1913, see page 25) was a Burroughs accounting machine which worked to fifteen decimal places and first-order differences. The National machine which was introduced by Comrie in 1934 and is in use to this day, works to thirteen decimal places and sixth-order differences. Even now no one has found it necessary to build a machine much more than half as big as the one which Babbage tried to make. His ambition to build immediately the largest difference

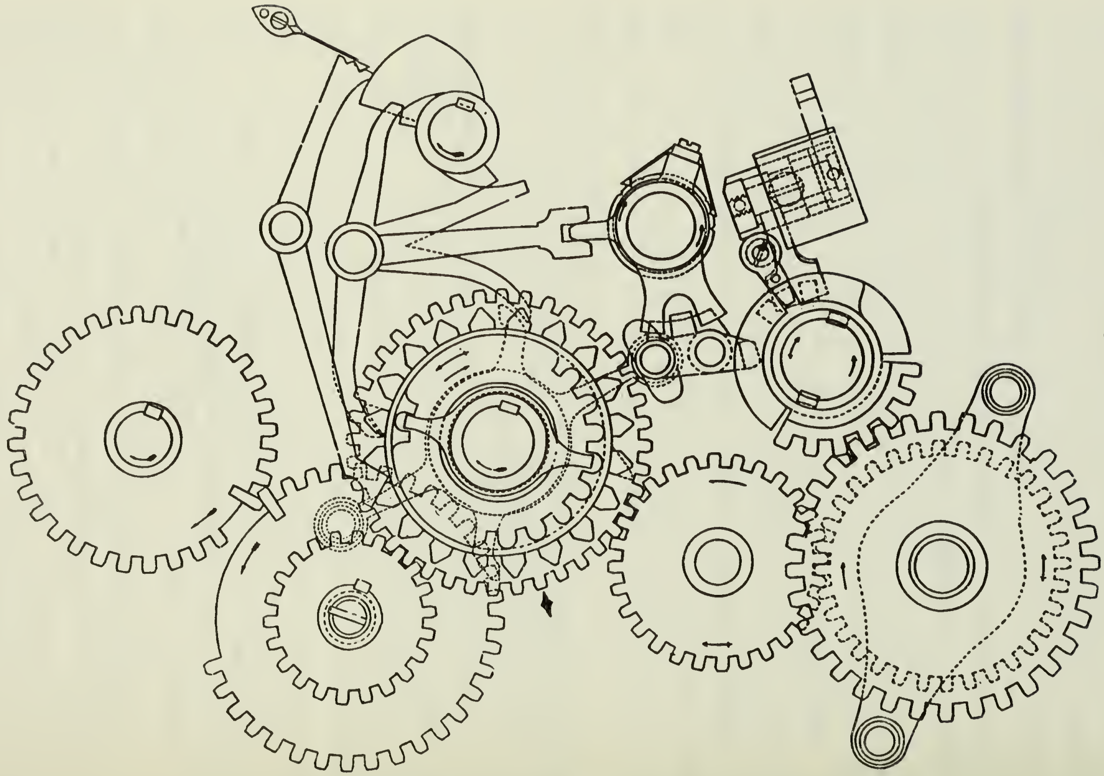


Fig. 1/1. One of Babbage's drawings of part of his engine

engine that could ever be needed probably delayed the exploitation of his own ideas for a century.

Once he had conceived the idea of his analytical engine, Babbage lost interest in the useful, simple, but less versatile difference engine and once again his plans were unnecessarily ambitious. For example, he planned to store 1,000 fifty-digit numbers. No machine which has so far been built has ever (so far as we know) had so many in use at one time.* The E.D.S.A.C., on which most of the electronic computing of this country has been done, has a capacity only one twentieth as big, it will hold about 250 ten-digit numbers in its delay lines. Babbage was compelled in the end to simplify his designs, but nevertheless had he ever finished his machine it would have weighed several tons, and contained thousands of parts all of which would have had to be precisely made and accurately fitted together. It is probable that even now it would be impossible to produce and operate an all-mechanical contrivance of such complexity. The problems which confronted Babbage can scarcely be imagined, for in a manner which is only too familiar to people who build computers today he found himself involved in a series of developments of great complexity which he had to study before he could begin to work on his main programme—he had to devise new techniques for machine drawing—the many thousands of drawings which he made for the difference engine were justly considered to contain some of the most remarkable draughtsmanship of the century (see Fig. 1/1). Babbage invented a new algebra with which to describe the movements of the interconnected parts of the machine—to evaluate their *logic* to use the modern phrase, and he himself accounted this algebra the greatest intellectual achievement of his life.

It is an ironic commentary on the fate of a visionary, that when he died, a disappointed man in 1871, his workshops contained what a contemporary account describes as "the wreckage of a brilliant and strenuous career . . . a machine which will probably remain forever a theoretical possibility," and Babbage was best known to his contemporaries as the inveterate enemy of street singers and barrel organists. After his death a committee of the British Association examined the Engine and reported: "Apart from the question of its saving labour in operations now possible, we think that the existence of an instrument of this kind would place within reach much which, if not actually impossible, has been too close to the limits of human endurance to be practically available." It is not

* Excepting, perhaps, in certain commercial calculations, see Chapter 22.

often that the report of a committee remains true after a hundred years without the alteration of a single word.

Babbage was obsessed by his plans for the machines for almost the whole of his life, and he had a lively sense of the magnitude of the task which confronted him. He failed to explain himself adequately even to his colleagues—in later life he was frequently and almost notoriously incoherent when he spoke in public. Even his friends with the best will in the world often found him hard to understand; he felt himself completely out of sympathy with his contemporaries, and, like many other inventors, both before and since, he was quite unable to cope with the Treasury. Let us remind ourselves that his successors in 1940 had not by any means understood or appreciated all that he did.

Shortly before his death he told a friend that he could not remember a single completely happy day in his life: "He spoke as if he hated mankind in general, Englishmen in particular, and the English Government and Organ Grinders most of all."

Poor man, he tried to solve by himself and with his own resources a series of problems which in the end taxed the united efforts of two generations of engineers. He was embittered by what he accounted his failure. Let us who can assess his achievements more accurately think rather of his foresight, his imagination and his courage.

Babbage wrote more than eighty books and papers and some rather discursive accounts of the analytical engine, but he said himself that he was too concerned with the development of the machine to write extensively about it. However, in 1840 he visited Turin and gave a series of lectures. These were attended by Menabrea (afterwards one of Garibaldi's generals), who was so interested in the subject that he wrote an account[(4)] of Babbage's ideas, and published it in the *Bibliothèque Universelle de Genève* in 1842. This paper was translated into English by the Countess of Lovelace, and published in *Taylor's Scientific Memoirs* (Volume 3).*

Lady Lovelace had undoubtedly a profound understanding of the principles of the machine, and she added greatly to the value of her translation by some comprehensive notes about the machine and a series of examples of its use, including what we should now call a *programme* for computing the Bernoulli numbers by a very sophisticated method. Babbage himself paid tribute to Lady Lovelace's paper, which he realized was by far the best contemporary account

* This paper has been out of print for many years. It is so interesting that it has been reproduced as an appendix to this book.

of his machine, and he remarked incidentally that the Countess corrected a mistake in his own analysis of the Bernoulli numbers.

Very little seems to be generally known about Ada Augusta, Countess of Lovelace, and I am much indebted to her granddaughter, the Rt. Hon. Lady Wentworth, who has helped me to prepare this brief account of her life. She was the only "daughter of the house and heart" of the poet Byron. The married life of Lord Byron, or rather the period during which Lord and Lady Byron lived together, was a year and a few days. They were married on 2nd January, 1815; on 10th December their only child was born, and in January, 1816, husband and wife separated for ever. When Ada was last seen by her father she was only a month old. It has been said of her that she did not resemble him in looks or in disposition, but at Crabbett Park there is a delightful portrait of her, dated 1832, which shows a distinct resemblance to Lord Byron. This portrait has been reproduced as the frontispiece by permission of Lady Wentworth.

Ada undoubtedly inherited her father's mental vigour. Her mother was both able and intellectual, and she studied mathematics eagerly up to the time of her marriage, under the tuition of William Frend, whose son-in-law, Augustus De Morgan, was later to teach her daughter. In a letter he wrote on 23rd December, 1812, Byron described her as the "Princess of Parallelograms" and it may have been her of whom he was thinking when he wrote—

'Tis pity learned virgins ever wed
With persons of no sort of education
For Gentlemen although well born and bred
Grow tired of scientific conversation.
I don't choose to say much upon this head,
I am a plain man and in a single station
But, Oh ye Lords of ladies intellectual,
Inform us truly, have they not hen-pecked you all?
(*Don Juan*)

It is certainly true of Ada Augusta that—like her father's Donna Inez—"Her favourite science was the Mathematical," and in this she was encouraged first by Lady Byron and then also by Lady Byron's intellectual friends—notably by Professor and Mrs. De Morgan, by Babbage, and by the celebrated Mrs. Somerville who was at the zenith of her career and reputation when Ada was brought up to town by her mother for her first London season. Mrs. Somerville describes in her memoirs how she did her best to foster Ada's mathematical talents and how together they "went frequently to see Mr. Babbage while he was making his Calculating Machines" and always found him "most amiable and patient in explaining the structure and use of the engines." Mrs. De Morgan also took Ada to see Babbage's machines and in her reminiscences has described one of the earliest visits in these words: "While the rest of the party gazed at this beautiful instrument with the same sort of expression and feeling that some savages are said to have shown on first seeing a looking glass or hearing a gun, Miss Byron, young as she was, understood its working and saw the great beauty of the invention." It is remarkable that the attitude of the average man to a computing machine should have changed so little in a hundred years.

Ada studied mathematics under Professor De Morgan for many years. He had an unbounded admiration for her abilities and her accomplishments; he went so far as to compare her with Maria Agnesi and he said of her that had she been a man, her "aptitude for grasping the strong points and the real difficulties of first principles would have lowered her chance of being senior wrangler, but would have made her into an original mathematical investigator, perhaps of first rate eminence."

Many of her letters are in the possession of Lady Wentworth, who has most kindly allowed me to inspect them. They show her to have had an interest in many branches of mathematics, and De Morgan himself said that many of her productions are of mathematical quality comparable with her translation of Menabrea's paper, which was written when she was 28. This paper shows her to have had a most remarkable understanding of Babbage's ideas, which she explained far more clearly than Babbage himself ever seems to have done.

Ada was a keen and accomplished musician, and played several instruments. In July, 1835, she married William, eighth Lord King, who was subsequently created first Earl of Lovelace; she had three children.

Both Lord and Lady Lovelace were very much interested in horses and in horse racing, and Lady Lovelace and Babbage eventually turned their attention to developing an "infallible" system for backing horses. Unfortunately, although their mathematics was sound enough, the partners failed to allow for the unpredictable vagaries of horses and their trainers. Babbage had a life-long tendency to devise quite impracticable schemes to raise funds for the construction of his analytical engine. After he had satisfied himself that the engine, when he had finished it, would be capable of playing a game of chess,* he designed a simple special machine to play noughts and crosses (tick-tack-too), and he proposed to build it and go on tour with it as a sort of variety turn. Perhaps he was inspired by the success of M. Maelzel's chess playing automaton, but he was dissuaded from these extreme measures by a kindly disposed member of the theatrical profession, who assured him that the British public had only quite limited funds with which to support enterprises of this kind, and that General Tom Thumb (the celebrated dwarf) had during the course of a recent tour collected everything that wasn't nailed down. Perhaps fortunately for himself, Babbage abandoned this idea, but the results of his interest in horse racing were far more serious. Both Lord and Lady Lovelace plunged heavily and lost, and Lady Lovelace went on betting after her husband had become disillusioned and stopped. This time she fell into the hands of some unscrupulous men, and had to pawn the family jewels, which were redeemed for her some years later by Lady Byron.

In 1852—still grievously harassed—she fell dangerously ill, and in November of that year she died at the age of 37 after a long, painful illness. By her own wish she was buried beside her father at Newstead.

A remarkable understanding both of mathematics and of horses seems to have persisted in Lady Lovelace's family for many generations, much as did pure mathematics among the Bernoullis or music among the Bachs. Her great grandfather, Sir Ralph Millbanke, owned a famous stud two hundred years ago; her daughter, Lady Anne Blunt, inherited all her mother's mathematical ability—some notebooks which were written when she was 19 show evidence of very great talent. She was an outstanding linguist, an artist and a musician (she owned two Stradivarius violins), but she devoted her life to the study of Arab horses and Arab tribes. She and her husband, the poet Wilfrid Scawen Blunt, travelled widely through the deserts of Arabia and Mesopotamia, where no European woman and few men had ever been before. Lady Anne became one of the foremost Arabic scholars of her day—she was consulted by the head of Azhar University on the interpretation of obscure Arabic classical texts. The Blunts discovered that the beautiful Arab horse was threatened with extinction in his native land, and so, at the request of old Mr. J. Wetherby, director of the General Racing Stud Book, they

* See Chapter 25.

imported some of the finest stock into England to preserve and improve the breed, with the idea of providing material for a possible reintroduction of Arab blood into the racehorse.* Wilfrid Blunt was famous among the Arabs and became "brother" to one of their chiefs; he was a magnificent horseman and a born leader of men—he could see Jupiter's satellites with his naked eye, but it was his wife's scholarly abilities and a million pounds of her money that were responsible for the success of the original Crabbett stud.

Since Lady Anne's death in Egypt during the 1914–18 war, her daughter, Lady Wentworth, has kept the Crabbett stud, which is now the most famous in the world. The Crown Prince of Arabia has said that there are no horses so fine in Arabia today. Crabbett horses have won races at record speeds over distances ranging from half a mile to more than 300 miles; all first prize winners in England in 1946, 1947, 1948, and 1949 were Crabbett bred, and of 5,100 entries in the American Arab stud book, more than 4,500 are wholly or partly descended from Blunt-Wentworth stock. Lady Wentworth herself has devoted many years to a study of the English thoroughbred horse, and has shown that with the exception of a few bars sinister of coarse blood it is descended entirely from Oriental blood imported into England about 250 years ago in horses the pedigrees of which were analysed by her mother in Arabia; she has shown quantitatively how all famous modern racehorses descend in direct male line from these few imported Arabs, and she has computed the fraction of the blood of each which is to be found in every Derby winner. The number of generations is now so great that her calculations have to be made to six significant figures. It is to numerical work of this type, as well as to her study of the conformation of her horses, that the success of the Crabbett stud is due.†

It was Lady Lovelace who made it possible for us to appreciate Babbage's genius, and we shall bring our story up to date by showing how modern versions of both the difference engine and the analytical

* "A good Arab stallion is the finest of all horses, a thousand times better than an English Thoroughbred for improving every breed." Napoleon Bonaparte.

† Lady Wentworth shows, for example that Bahram, who won the Derby in 1935, has 222,281 crosses to the Arab mare Old Bald Peg, and to her sire Sultan; 123,037 to the Darcy White Turk; 179,105 to the Darcy Yellow Turk; 112,667 to the Leedes Arabian; 64,032 to the Byerley Turk; 44,079 to the Darley Arabian; 28,232 to the Godolphin Arabian; 66,893 to the Helmsley Turk; and 40,305 to the Lister Turk. (Turk was another name for Arabs imported through Turkey, which for a hundred years included Mesopotamia and North Arabia.) This is a remarkable application of binary arithmetic (see page 33).

engine have been exploited, the first in 1913, the second about thirty years later. It will be necessary to describe these two developments quite separately, but we must first remark that modern achievements in electronic engineering have made it possible to realize Babbage's ideas in a form he could not possibly have foreseen. His machines were to have been all mechanical, and his drawings are hard to understand (see Fig. 1/1). We shall therefore not attempt to describe them in detail, but we shall explain something of the methods by which in recent years the principles which Babbage enunciated have been embodied in many enormous computing machines which have been built in all parts of the world.

First of all, however, we must pursue a very interesting offshoot of Babbage's work, which is well known and of great importance in modern business. He had to find a mechanical method of controlling the operations of his engine and of presenting numbers to it in a form it could understand. He found the answer in the *punched cards* which had just been invented by Jacquard to control his looms. When a pattern is woven into a cloth it is necessary to lift the threads of the warp in proper sequence, and to control simultaneously the throw of the shuttles which carry the weft. Jacquard arranged to fasten all the threads which move together to a single rod, and to move the rods in turn by cards which pushed against all of them and lifted those which did not pass through a set of holes punched in the cards. A repetitive pattern could be woven by using the same batch of cards several times in succession, but 24,000 cards were needed to produce a famous portrait of M. Jacquard himself. Once when the Prince Consort came to see Babbage he saw this portrait, which was hanging on the wall of the drawing room. The Prince recognized it for what it was (most people mistook it for an engraving) and Babbage was much impressed by the fact that he was immediately able to appreciate the fact that the technique which had been used to weave it might be taken over and used in a computing machine.

Twenty years after Babbage's death Dr. Hollerith, who was at that time in charge of the American Bureau of the Census, despaired of reducing and tabulating the data for which he was responsible. He estimated that it would take twenty years to complete the work he had to do on each census, and the law required that a census should be taken every ten years.

History may now be repeating itself. The British Government takes a Census of Production every year. The forms for the 1951 census were sent out before the end of 1951, but the results of the

1948 census were not complete and only a preliminary analysis of the data had been published in August, 1951. Information which is published so belatedly may prove to be invaluable to some future Gibbon chronicling the decline and fall of British trade, but it has lost much of its value to those responsible for Government policy before it becomes available to them. It may be possible to solve some of the problems of handling masses of data of this kind by the use of modern digital computers (see Chapter 21). Dr. Hollerith solved his problems by inventing and introducing methods of recording data by punching holes in cards, and also a series of machines for sorting cards and analysing the data they contain. Since his time the International Business Machines Corporation (I.B.M.) in America, and the British Tabulating Machine Company, have developed his ideas, and the Powers-Samas and Remington Rand Companies, ideas of his assistant Mr. Powers. In France the *Compagnie Bull* exploits patents due to a Norwegian who left them to Oslo Hospital when he died. All these companies have developed complicated machines for processing data which have been punched on cards, and systems of accountancy based upon these machines are in widespread use by progressive firms all over the world.

Punched-card equipment has not solved all the problems of the Census however. At the request of U.N.E.S.C.O., the Siamese Government took a population census in 1947. They were equipped with a large battery of I.B.M. machines, but for some reason it appeared that the results of the analysis were very slow to emerge, and as none were available eighteen months after the sorting process had begun, an investigation was made by a group of punch-card experts. They found the "bottle-neck" quite easily. The Siamese distrusted the card machines, and right at the end of the line, they had one Chinese girl who was supposed to be checking all the results with an abacus. The poor girl was running seventeen months behind schedule and meanwhile there seemed to be a chance that white ants would eat the cards.

We may anticipate current developments by remarking that many modern computing machines use punched cards in the manner which Babbage proposed both to control his engine and to feed it with data. Plate III shows a photograph of a couple of cards punched for the A.C.E. at the National Physical Laboratory in Teddington.

Our story would be incomplete if we failed to pay tribute to the work of another pioneer in the field of computation, the late Dr. L. J. Comrie, who followed up the pioneer work of T. C. Hudson, (references 16, 17 and 18). Long after the foundation by Charles II of the Royal Observatory at Greenwich there were still no really satisfactory tables of the positions of the sun and the stars, and it was not until 1766 that Maskelyne, who was then Astronomer Royal, published the first *Nautical Almanac*. This work has been published every year since then, and by using it a sailor equipped with sextant and chronometer can determine his position at sea.

The labour which is involved in computing astronomical data of this kind is very great, and for many years was undertaken by highly skilled computers, most of them elderly Cornish clergymen, who lived on seven figure logarithms, did all their work by hand, and were only too apt to make mistakes.

The first edition of the *Nautical Almanac* was prepared with all the precautions that were known at the time. Nevertheless there were hundreds of mistakes in the astronomical data from which it was computed, and a single individual found more than a thousand errors due to faulty computation. It is difficult to imagine the temerity it required to navigate a ship on the high seas with an unreliable chronometer, inaccurate log-tables and an untrustworthy almanac. We have already seen what Babbage thought about this situation, and what little success he had in trying to remedy it.

Improved techniques in hand computation and greater experience had already eliminated errors from all navigational tables, but it was not until 1926 that the work of mechanizing the calculations was begun by Dr. Comrie, who was then Deputy Superintendent of the Nautical Almanac Office. Today no logarithms are needed. About one-third of the staff are astronomers, and the rest computers, many of them young girls who can successfully operate their desk calculating machines but who may be quite unable to explain the complicated functions which they are computing. The complete mechanization of these computations cannot be long delayed.

The reader will perhaps be surprised and horrified to learn that the calculations for the *Nautical Almanac* are neither duplicated nor repeated; however, their accuracy is of such vital importance that they must be checked somehow. This is done by the method of differences, which we have already explained. Babbage's difference engine which mechanized the process had been in the Science Museum since the nineties, and no further progress was made until Comrie discovered that the Burroughs accounting machine, which had been designed and built in America and was intended for purely commercial applications, could be used without modification as a difference engine. Since then this has been exploited extensively for the computation of mathematical tables and for scientific calculations of all kinds. For example, in 1931 a single machine printed out 30,000,000 figures, which is more than a copyist could copy, let alone compute, in seven years.

The first scientific application of the Hollerith system was undertaken by Dr. Comrie in the Nautical Almanac Office in 1928. Information taken from E. W. Brown's *Tables of the Moon*, punched in the form of twenty million holes in half a million cards, was used to compute the position of the moon at every noon and midnight from 1935 to A.D. 2000. Something like a hundred million figures were added in groups and the results printed in the course of seven months. From the results the tables in the *Nautical Almanac* which give the position of the moon every hour were prepared by the difference engine, in its modern form—the National accounting machine.

Professor Brown had begun to do the work by hand and Comrie often recalled the "ecstasies of rapture" with which Brown watched the addition of his own figures at the rate of 20 or 30 a second. The enthusiasm with which Professor Brown described the process on his return to America probably stimulated W. J. Eckert, the leading American pioneer, in the application of these machines to scientific calculations (see page 285).

Comrie's achievement was to exploit the fact that standard computing machinery, originally devised for commercial accountancy, could be adapted for scientific work, and that it fulfilled the functions of Babbage's difference engine, and, in part, of his analytical engine. We shall now have to analyse the art of computation in more detail in order to understand the profound importance of the ideas which Babbage proposed to incorporate in his analytical engine. We shall conclude that the modern versions of this machine—the high-speed universal automatic digital computers, which have been developed for scientific work—may not only have a profound effect in this field, but may in their turn revolutionize commercial accountancy.

The Problem of Computation

A human computer working at his desk needs a calculating machine, reference books of tables, pen and paper with which to record the intermediate results in his calculations, and instructions as to how to proceed. This is not all; computing is something of an art, and the computer will be inefficient, and may not get very far, unless he has some power of discrimination which enables him to interpret his instructions in the light of the results which his computations have produced, and, if need be, to modify his procedure accordingly. The power of discrimination of which a human operator is capable cannot be exercised automatically by any of the machines which we have discussed so far. Their potentialities are therefore limited. Babbage clearly understood the restrictions imposed by the inability of a machine to make decisions for itself. He was able to take the next step and to suggest how to endow a machine with the minimum amount of "intelligence" which it needs. As he expressed it, he made the machine "bite its own tail." It is entirely due to these ideas of his that the modern computing machines to which we shall devote the rest of this book are so fast, so flexible and capable of such an astonishing variety of operations.

If a machine is to perform the functions of a human computer, it must possess—

(*a*) An arithmetic unit, capable of performing the normal operations of arithmetic. Babbage called this unit the *mill.*

(*b*) A memory, that is to say a mechanism which will retain numbers which are needed in the calculation and also the instructions which will be needed to define successive stages in the computation. Babbage called this part the *store.* He planned to store 1,000 numbers, each of 50 decimal digits. The new Manchester machine stores three times as many, but most machines which have been built have a much smaller capacity.

(*c*) A built-in power of judgment, which will enable the machine to choose, according to prescribed criteria, the course which the computation has to take.

(*d*) An input-output mechanism which allows the operator to feed numbers and "instructions" into the machine, and to extract from it the results of a calculation. As we have already stated, Babbage planned to use punched cards for input. He was going to use them as an auxiliary store, for example for tables of functions. Most modern machines make use of standard teleprinter tape, but some (for example the A.C.E., see Chapter 8) use Hollerith punch cards. Babbage also proposed to use punched cards as one form of output, but realizing as he did the risk of error in copying tables by hand, he proposed to make the machine set up its results in type where necessary. Modern machines use both tape and card for input and output and print out their results automatically as well. The sheets so prepared can then be reproduced without error by photo-lithography.

Subsequent chapters of this book contain detailed accounts of several machines and of the components from which they were built, but first of all let us consider how a machine can be made to satisfy the conditions which we have just enunciated.

The nucleus of all machines is the store or memory in which information can be kept. Several types of memory will be described later on; for the moment it suffices to say that some machines use sound waves in mercury, others use tiny magnets on the surface of a rotating drum (much like those needed to store music in a tape recorder). Babbage was going to use holes in cards, and metal discs which would turn round on spindles. The Manchester machine uses patterns of electric charge arranged in lines on the surface of a cathode ray tube. Each line contains a number, or else an instruction. The correspondence between the patterns of charge and the information which they convey need not concern us for the moment. The lines on the cathode ray tubes are encountered consecutively, and the serial number of each line is called its *address*. An instruction consists of two parts—an *address* part, and a *function* part. When the machine reads an instruction it re-arranges its own internal connexions as a telephone exchange does, and then reads off the number whose address has been specified, and sends it into that part of the mill (or arithmetic unit) which is appropriate. For example, an instruction such as 137A might mean: "add the contents of line number 137 into the accumulator store." The next instruction might be 106B which would mean: "put the contents of the accumulator into line 106 in the main store." The time taken by a machine to read an instruction and to obey it varies very much from machine to machine and in fact the central problem in machine design is to produce a memory the contents of which can be read out very quickly. The Manchester machine and the E.D.S.A.C. will both obey about 1,000 instructions a second. Babbage's machine was to have done 60 additions per minute or one multiplication of two fifty-digit numbers per minute.

Every machine contains a central control unit which under normal circumstances causes it to read out the instructions in the serial order in which they appear, but among the instructions are some which react upon the control unit, and cause the machine to jump to some other instruction in the store. For example, it is possible to make the machine go back to a previous instruction and repeat a particular series of instructions several times over if necessary. Since the instructions are kept in the same form as numbers and in

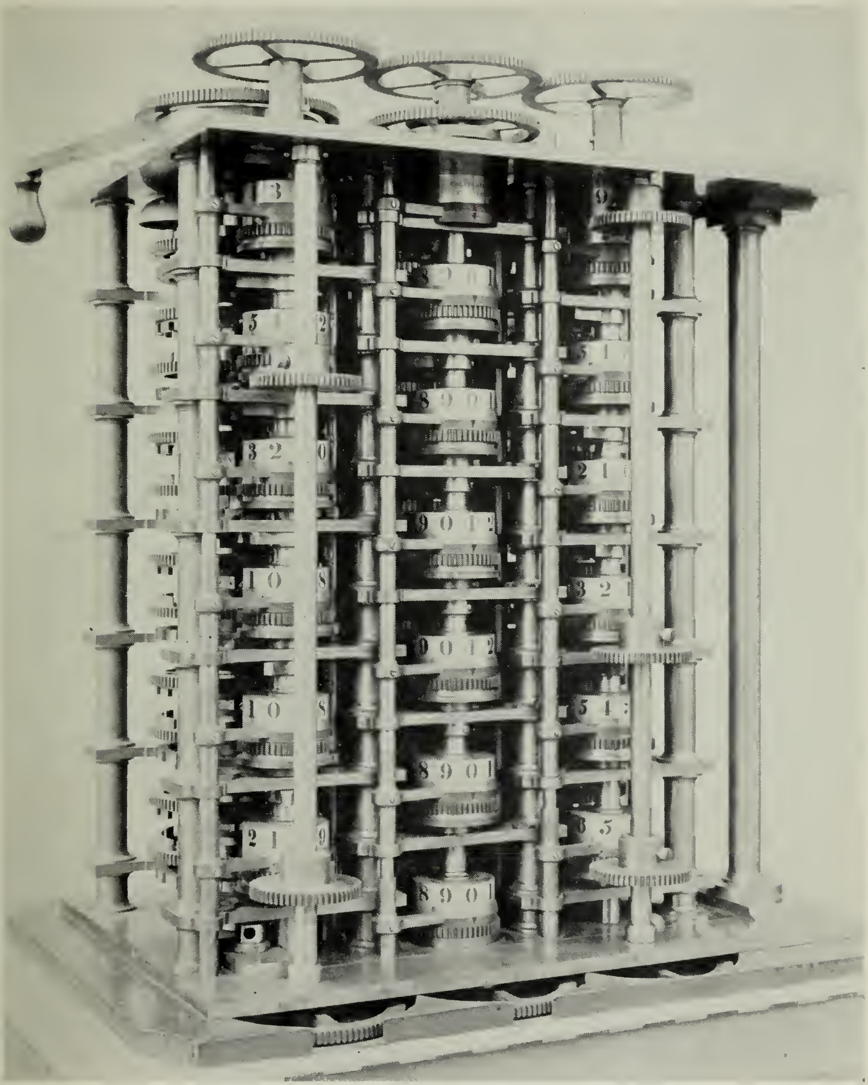


Plate II. Part of Babbage's Difference Engine
From an exhibit in the Science Museum, South Kensington
(Crown Copyright Reserved)

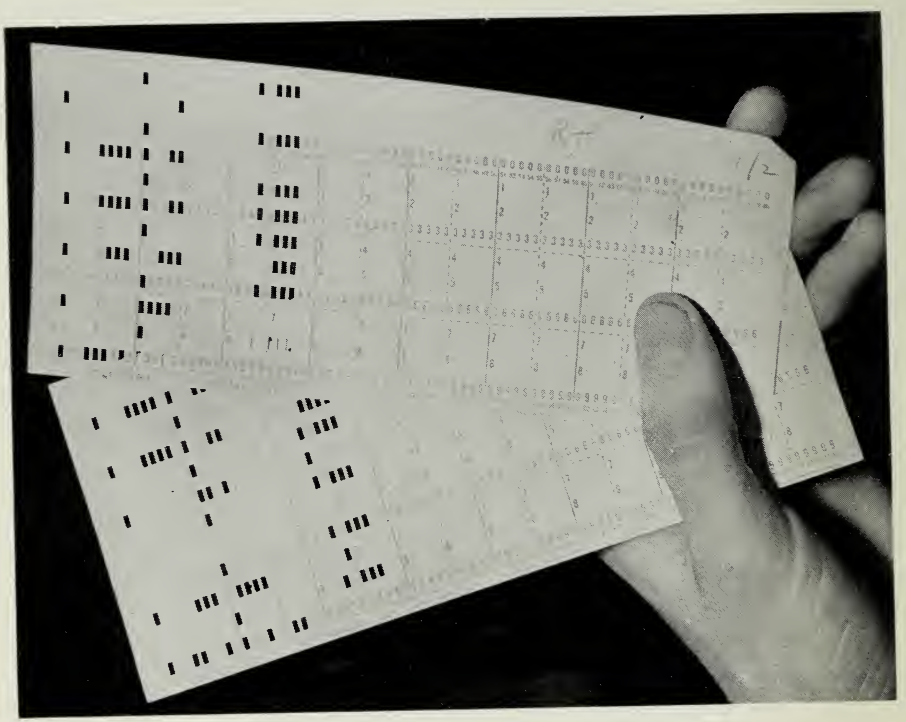


PLATE III. TWO HOLLERITH PUNCH CARDS OF THE TYPE USED IN THE A.C.E.

the same store, it is possible to operate on them in the mill. In particular it is possible to modify the address which they contain by a specified amount. For example, the instruction 137A can be altered by the machine itself to 138A. This means that the machine can be made to follow out automatically the same series of arithmetic operations on several different numbers in succession. If every single operation which the machine is to perform had to be separately written out for it by a mathematician there would be no point at all in making it work so fast. As it is the operator can prepare a single routine which the machine can follow hundreds of times.

Suppose that the store contains a table of some kind; successive entries in which are stored in lines 500 to 600—a table of logarithms for instance. If the machine has to look up log 52 (it may have computed the number 52 itself) it adds 52 to the constant number 500, and adds the total to an instruction (o I). Automatically the logarithm of 52 will go into the mill. In fact the machine can calculate so quickly that one would never store logarithms in it, but would work them out as one needed them; nevertheless some tables do have to be stored and consulted in this way.

We have so far ignored the most important of these "transfers of control," and that is the one which the machine decides upon for itself. It can be instructed to choose the next instruction which it is to obey according to the sign of the number in the accumulator. The mathematician need not know at what stage in his calculation the sign will change, but he instructs the machine what to do *if* it changes. He might, for example, arrange for the machine to follow out the same series of instructions on different numbers until the sign changes, and then to carry on with the next step in the computation.

At this point it is necessary to interpolate a remark to explain our apparent attribution to the machine of certain human qualities, such as *Memory*, *Judgment* and so on. Modern digital computers are capable of performing long and elaborate computations; they can retain numbers which have been presented to them or which they have themselves derived during the course of the computations; they are, moreover, capable of modifying their own programmes in the light of results which they have already derived. All these are operations which are usually performed (much more slowly and inaccurately) by human beings; but it is important to note that we do not claim that the machines can think for themselves. This is precisely what they cannot do. All the thinking has to be done for them in advance by the mathematician who planned their programme and they can do only what is demanded of them; even if he leaves the choice between two courses of action to be made by the machines, he instructs them in detail how to make their choice. The abilities and limitations of a machine were foreseen by Lady Lovelace, who wrote in 1842: "The Analytical Engine has no pretensions whatever to originate anything. *It can do whatever we know how to order it to perform.*" (See page 398.)

We shall discuss later some of the philosophical implications of these ideas; here it suffices to remark that the metaphorical attribution of human qualities to machines is commonplace and convenient in certain circumstances and it is usually understood by everybody. These new machines have so many of the qualities of a human brain that one has to be rather careful if one is to avoid giving, by implication, a misleading impression. For convenience, and to avoid circumlocution, we shall frequently make use of words which are usually applied only to the processes of thought in a human brain, but we are not trying to support any particular theory of the construction and operation of the brain itself. Many years ago, Babbage tried to use his machine to illustrate some of the major problems of theology, and in much the same way as Sir James Jeans seemed to think of the Creator as a Mathematician, Babbage seems to have thought of Him as a Programmer.

We shall confine ourselves in this book to an account of the machine and its applications; we shall not attempt to consider any part of theology as an experimental science, and we hope thereby to avoid laying ourselves open to charges of anthropomorphism.

We shall now interrupt the thread of our story, and explain how a typical machine is made, what happens inside it and how it is controlled. After we have explained how we endow the machine with the power of choice we shall proceed to discuss how a mathematician can exploit it to perform the most intricate calculations that one can imagine.

The outstanding problems involved in making machines are almost all technological rather than mathematical. Babbage's ideas on the mathematical design and the logic of computer construction are astonishingly modern, but he was quite unable to build one in spite of a lifetime's efforts; we have had to wait a hundred years for the fulfilment of his plans. The preparation of the detailed design on paper calls for most subtle and sophisticated reasoning, and at the present time it can be done efficiently by a very few people, but it represents only a small part of the whole effort which is necessary to produce a working machine. The performance of the finished product will depend on the work of many engineers, some of whom may never have heard either of the art of computation or of mathematical logic.

REFERENCES

1. *Barlow's Tables of Squares, etc.* (1814).
2. *Dictionary of National Biography*, "Charles Babbage."
3. *Babbage's Calculating Engines.* A collection of papers relating to their history and construction, edited by Major-General H. P. Babbage (1888). This book contains a list of all Babbage's published papers.
4. Article xxix from *Taylor's Scientific Memoirs*, Vol. III (*see* Appendix).
5. Obituary notice of Charles Babbage in *The Gentleman's Magazine*, **39**.
6. Babbage, C. *Passages from the Life of a Philosopher.*
7. Babbage, C. *The Ninth Bridgewater Treatise.*
8. The obituary notice of Lady Lovelace in *The Gentleman's Magazine*, 1852.
9. A note which was written by Mary, Countess of Lovelace, for Mayne (Miss). *The Life of Lady Byron*, Constable (1929).
10. *The Life of Augustus de Morgan*, by his wife.
11. Somerville, Martha. *Personal Recollections of Mary Somerville* (by her daughter). (London: 1873.)
12. "The writings of William of Malmesbury (Gista Reg. Angl.)" as translated by R. Allen. *British Historical Review* (1892).
13. Babbage, C. *The Decline of Science in England.*
14. Wentworth, Rt. Hon. Lady. *Thoroughbred Racing Stock and its Ancestry.*
15. Wentworth, Rt. Hon. Lady. *The Authentic Arabian Horse and its Descendants.*
16. *Handbook of the Napier Tercentenary Exhibition* (1914).
17. *Monthly Notices of the Royal Astronomical Society* (Apr., 1932).
18. *Journal of the Royal Statistical Society*, Supplement to Vol. 3, No. 2 (1936).

Chapter 2

THE CIRCUIT COMPONENTS OF DIGITAL COMPUTERS

"*. . . What* is *the use of repeating all that stuff*," *the Mock Turtle interrupted*, "*if you don't explain it as you go on? It's by far the most confusing thing I ever heard!*"—Lewis Carroll*

Although digital computers of the type described in this book appear to be very complicated and difficult to understand, it is not hard to work steadily through a description of any particular machine, step by step. It might appear impossible for anyone but an expert electronic engineer to understand the electronic circuits from which the machine is made—but this is not true. A digital computer may contain several thousand valves and other components, nevertheless all these valves are used in a few simple types of circuit, each of which can be understood quite easily. It is the purpose of this chapter to explain the operation of the circuits, and of other special components which are needed. In the next chapter we shall show how these "bricks" are assembled to make a machine.

Representation of Numbers and Information in the Machine

It is very convenient in electrical circuits to make use of devices which have only two "states," for example, a relay or a switch may be open or closed, but if it is working properly it will never be half on and half off; a spot may or may not be present on the surface of a cathode ray tube, a pulse may or may not pass through a circuit during a certain interval of time; a hole may or may not be punched in a certain place in a piece of teleprinter tape. It is easy to use electronic valves as relays which are *on* or *off*—by comparison it is much more difficult to arrange, for example, that a valve is turned exactly one-, two-, or three-tenths *on*.

Even the most primitive civilizations seem to have understood[(1)] that any type of information can be expressed in a *two-symbol* code; for example, the bush telegraph signals of many African tribes use drum beats with high and low pitch.[(2)] The ordinary Morse code of

* And unless the reader has at least a nodding acquaintance with electronics, he is advised to skip the greater part of this chapter. The rest of the book is comprehensible without it.

"dots" and "dashes" may be less sophisticated than the code of the drums, but it is probably better known to the reader. The two codes resemble each other in that both of them use two signal elements with which to transmit intelligence, and in neither of them does the mere absence of signal convey any information. In this respect these two codes are more like that used in the T.R.E. machine (see page 144) than any other described in this book.

Our conventional decimal system was developed more than a thousand years ago by the Hindus from methods of counting which had been suggested by the number of digits on the two hands. It is quite unnecessary, when we are making a computing machine, to handicap ourselves in an attempt to comply with a tradition which was originally suggested by the fingers of our ancestors. The *binary system* is better suited to the anatomy of the machine.*

This binary system which most machines employ makes use of only two symbols, namely 0 and 1. Table II shows a few numbers in the ordinary decimal notation, and also in the equivalent binary form. The number 137, by which of course we mean 1×100 plus 3×10 plus 7, becomes $128 + 8 + 1$ (i.e. $2^7 + 2^3 + 2^0$) or 10001001 in binary form.

TABLE II

BINARY EQUIVALENTS

Decimal	Binary
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010
100	1100100
1000	1111101000
8	1000
16	10000
32	100000
64	1000000
128	10000000

* It is said that Binary numbers were known to the Chinese Won-Sang (1182–1135 B.C.) who wrote *The Book of Permutations*, the third in antiquity of the Five Canons of the Chinese. They were advocated by Leibnitz, and they were used for the first time in fast electronic counting circuits by Wynn-Williams in 1932.

An ordinary desk machine which will handle six-digit numbers represents *minus one* as 999999. In the same way a twenty-digit binary machine represents minus one as 11111 11111 11111 11111, and arithmetic circuits have to be suitably designed to handle negative numbers written in this way. It is particularly important to note that all negative numbers have a 1 as their most significant digit, and the convention which is adopted is that all positive numbers start with 0. The sign of a number is indicated by its most significant digit.

In spite of the fact that it requires about $3N$ binary digits to express a number of N decimal digits, the construction of a machine to work in binaries is, at any event at this stage of the art, rather simpler than that of an all-decimal machine. A few American machines, and in this country the Harwell and the R.A.E. machine (see Chapters 9 and 12) work throughout in decimals, but the majority of machines now being built operate on numbers in binary form. The conversion of numbers from one system to the other is a perfectly straightforward arithmetic process, and can be carried out by the machine itself. The speed with which modern machines work is such that (for example) the E.D.S.A.C. converts an eight-digit decimal number into binary form in less than one-tenth of a second; the converse process takes about as long, so that the time taken for the two-way conversion is negligible, and in fact the mathematician who is using the machine, and will, of course, be using decimals, need never be aware of the fact that it is taking place, and that the machine is handling his numbers as binaries.

Information of all kinds, including both the instructions which the machine is to obey and the numbers with which it is operating, is handled in most machines in the form of trains of pulses. It is necessary to distinguish (for example) between the pulse which represents the number 2^0 from that which represents the number 2^3 and this can be done in two ways. We can arrange that the significance of a pulse is determined by the wire on which it travels, or by the instant at which it occurs. This distinction is quite fundamental, and leads to the existence of two types of machine, usually known as *parallel* and *series* (or serial) respectively. In a parallel machine all the digits of a forty-digit number arrive simultaneously and if two such numbers have to be added all the digits have to be dealt with at once (by 39 adders). In a serial machine the digits are dealt with in succession and so two chains of pulses can be added by a single electronic adder. It is roughly true to say that parallel machines are larger, faster and more complicated than serial machines—they seem too to have a larger Hartree Constant.

Most of the machines which are now working are of the serial type, but parallel machines have been under construction for several years. Examples of both types of machine are found in this book. Many of the circuits which are described in this chapter could be used in machines of either type, but greater emphasis has been given to serial operation, since this has (so far) been more widely used in this country.

Information is transmitted through our own bodies by chains of pulses all of which are about the same size. The intensity of a stimulus controls the number of pulses which pass through a nerve fibre in a second, but their amplitude (about a tenth of a volt) and their velocity (about 200 miles an hour) are more or less constant. The distinction between serial and parallel operation is also clear in the nervous system. The ear hears simultaneously all the frequencies which are presented to it, the eye can only see distinctly a very small area at one time, and has to "scan" the field of view piece by piece and to build up a complete picture in serial form. An all-parallel eye might be more efficient but it would be impossibly complicated and too big to carry about.

Since it is the presence or absence of a pulse in a particular interval of time which is important in a serial machine, it is necessary to provide a time scale or "clock." In the Manchester Mark I machine for example, a 100 kc/s quartz oscillator runs continuously, and provides standard time for the whole machine; all wave forms are derived from this crystal. First of all a series of "pips" 10 μsec apart is produced, these pips are fed into 24 pairs of valves connected in a chain. This circuit acts like a high-speed rotary switch connecting 24 wires in succession to a source of potential. Each wire carries a pulse every 24×10 ($= 240$) μsec. These pulses, which are called the *P* pulses, are taken all over the machine and the significance of each one of a chain of pulses representing a number or an instruction is determined by the particular *P* pulse during which it occurs. All the pulses in a chain are of fixed amplitude and duration. They can only be present or absent; if they are present, their size, duration and timing are all standardized and determined by the machine. The path which the pulses take through the machine is determined by a series of electronic "gates" which behave very much like very high-speed relays of the normal type (they are described in detail on pages 43–6). The instructions control the gates, and information passes through them in such a way that the successive operations which are performed are those required by the programme. There is here a fairly close analogy with an ordinary telephone exchange. By dialling a number a subscriber sends successive pulse trains to the exchange. These pulses, which act as "instructions," open or close several hundred relays and establish a connecting path for the subscriber from one part of the exchange to another. He can then transmit further information through this path. If he were to use a teletype machine, the signals which he transmitted would be similar to those which controlled the operation of the exchange itself. Now, a telephone exchange is complicated, not so much because of the complexity of the components from which it is built, but because of the complexity and multiplicity of its interconnexions. Computers are complicated in much the same way and for much the same reason. An attempt to explain in detail how they work will not be made, but a description of principle and a simple account of some of the novel circuit elements will be given.

Applications of the Principles of Mathematical Logic to Digital Computers

Later chapters of this book contain an account of the way in which computing machines may be used to investigate problems in logic, as well as doing numerical and mathematical work, but it is necessary here to anticipate very briefly some of the sequel, and to put the reader in the picture by explaining something of the two subjects of *Boolean algebra* and *mathematical logic*, for they have to be used in the design of the machines. George Boole (1815–64) is the classical case of the Horatio-Alger hero who learnt mathematics. He had a poverty stricken childhood, but ultimately he became professor of mathematics in the University of Cork, where he published papers on the laws of thought. He is almost the only mathematician of first-rate eminence whose first papers were written when he was more than forty years of age. His ideas have since become of great importance to professional mathematicians, but they are still almost unknown to the ordinary layman.

The ordinary algebra with which all schoolboys become familiar will handle problems of this type: "If a number is multiplied by itself, and the resultant is added to four times the number itself, the product is 21. What is the number?" Boolean algebra is designed to handle questions such as: "A club has the following rules; (*a*) The Financial Committee must be chosen from among the General

Committee. (*b*) No one shall be a member of the General and Library Committees unless he is also on the Financial Committee. (*c*) No member of the Library Committee shall be on the Financial Committee. Simplify these rules." Both these problems can, of course, be solved without the use of any algebra at all, but everyone knows that the symbols and methods which have been evolved for handling the first type of problem will allow a second-rate mathematician who is familiar with them to solve problems which completely baffled the ancients who tried to handle them. It so happens that Boolean algebra is not taught in schools so far, though in its simpler forms it is not particularly hard to do, and its symbolism is no more complicated than that of ordinary algebra. Moreover a study of the rules of any insurance company, or of the rules which are issued by a welfare state, such as for example, the 20,000-word circular in which the O.P.A. described the regulations which govern the sale of cabbages, might suggest that an ordinary citizen is more likely to have occasion to use Boolean algebra than classical algebra in the course of an ordinary lifetime.

It is to be noted that neither algebra will handle problems such as: "If a boy aged twenty can gather ten pounds of blackberries in a day, and a girl aged eighteen can gather nine pounds, how many will they gather together?" The answer is, of course, that it all depends, but probably not very many. If the reader feels himself to be affronted by the absurdity of such a problem, let us remark quite firmly that it is at least as representative as either of the other two examples of the questions which arise in everyday life. Its solution depends entirely on information which may never have been explicitly stated, but which is probably well known to a practical farmer. It is precisely because a mathematician may be carried away by the beauty of a complicated chain of reasoning, which ignores essential and familiar data, that he gets a reputation for unworldliness, and that an expert is sometimes defined as one who solves minor problems and avoids small errors as he sweeps forward to the "Grand Fallacy." It is for this reason among others that it is so hard to do commercial work on a computing machine; this difficulty, of making a machine take an overall view of a problem, of ensuring in fact that not only is its arithmetic impeccable, but that its results have some meaning in a real world, is always extremely hard to resolve. It was deliberately introduced in rather a light-hearted way, but it will recur again and again in this book, especially in Chapters 22, 25 and 26. But to return to Boolean algebra.

It is necessary first of all to explain the nomenclature which is conventionally used when discussing the properties of the electronic relays or *gate circuits* as they are called, for they are of major importance to the designer, and illustrate a surprising connexion between the design of digital computers and these unfamiliar branches of mathematics.

Two types of gate which are used in large numbers are called *And* gates and *Or* gates. These names are derived from the operations in mathematical logic called *And* and *Or*.

In mathematical logic, a proposition is a statement which is either true or false, without any ambiguity. Two propositions, *A* and *B*, can be combined in several different ways to produce a third, *C*;* for instance, we may have the following relationship—

C is true when either *A* is true *or B* is true. This is the analogue of the *or* gate, which produces an output pulse when there is a pulse on either *A* or *B* (the two inputs). This relationship can be put into arithmetical form if we allow 1 to signify truth and 0 falsity. Thus *C* is 1 when *A* is 1, or *B* is 1; which implies that *C* is 0 if and only if *A* and *B* are both 0.

We may interpret the relation *or* as a kind of addition in which the "logical sum" of 0 and 0 is 0; that of 0 and 1 is 1, and that of 1 and 1 is also 1. The logical *or* relationship is sometimes called *disjunction*, and is represented by the symbol v. In this notation—

$$0 \vee 0 = 0$$

$$0 \vee 1 = 1 \vee 0 = 1$$

$$1 \vee 1 = 1$$

Similar considerations apply to the relation *and*, which is usually expressed by the symbol &, or simply by a dot. The *and* relation is sometimes called *conjunction*, or sometimes *collation*. In the language of logic, *C* = *A* and *B*, means that *C* is true if, and only if, both *A* and *B* are true. In the language of gates, the output *C* of an *and* gate carries a pulse if, and only if, both inputs *A* and *B* carry pulses. In the language of logical arithmetic—

$$0 \& 0 = 0$$

$$0 \& 1 = 1 \& 0 = 0$$

$$1 \& 1 = 1$$

Replacing the ampersand by a dot, it becomes obvious that conjunction (collation) has the properties of multiplication.

* See also Chapter 15.

There are other logical operations which have their counterpart in electronic circuits as well as in arithmetic. Only one of them will be mentioned here. Let C be derived from A and B according to the following rule: C is true either if A is true and B false; or if A is false and B true. But C is false if A and B are either both true or both false. Under these conditions, C is called "A not-equivalent B," indicating that the truth of C depends on the truth-values of A and B being different. We shall not attempt to reformulate this relation in terms of circuitry, nor to show a relay or gate-circuit realizing it—but it is interesting to note the arithmetical form—

$$\begin{aligned} 0 \not\equiv 0 &= 0 \\ 1 \not\equiv 0 &= 1 \\ 0 \not\equiv 1 &= 1 \\ 1 \not\equiv 1 &= 0 \end{aligned}$$

This process can be described as *addition modulo* 2; it has the same structure as the relation—

$$\begin{aligned} \text{even} + \text{even} &= \text{even} \\ \text{even} + \text{odd} &= \text{odd} \\ \text{odd} + \text{even} &= \text{odd} \\ \text{odd} + \text{odd} &= \text{even} \end{aligned}$$

We have mentioned this rather out-of-the-way subject of mathematical logic for various reasons. Firstly, *or* and *and* gates form the basis of most of the circuitry of the machine. Secondly, the arithmetic operations of the machine can be analysed in terms of the fundamental logical operations; this analysis in turn leads to the design of the circuits accomplishing these operations. Again, some of the actual instructions cause the machine to perform the operation of disjunction, conjunction (collation) and non-equivalence (addition modulo 2) on all the digits of two numbers simultaneously, and these instructions have turned out to be extremely useful in the construction of programmes. Finally we find it fascinating to think how the abstruse mathematical speculation of philosophers like Boole and Bertrand Russell, who have shown how to base arithmetic on logic, has its counterpart in electronic hardware of practical importance.

INTERLUDE ON CIRCUITRY

In order to describe the electronic circuits which make up a computer it is necessary to explain the working of two types of radio valve, called the *diode* and the *triode*.

Both the diode and the triode are circuit elements which can pass electric current in one direction only, and the two types of valve differ merely in the method used for controlling that current.

We shall consider first the diode, which, as its name implies, has only two electrodes, the anode and the cathode, and which is the simplest type of valve to be made. The cathode is hot, and emits

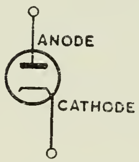
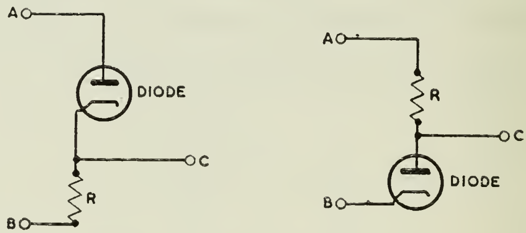


Fig. 2/1. The symbol for a diode



(a) (b)

Fig. 2/2. Simple circuits containing diodes

negative electrons which will cross the evacuated space and fall on the anode, if this electrode is at a more positive potential than the cathode from which the electrons came. The symbol for a diode is shown in Fig. 2/1.

Suppose we consider a simple circuit as in Fig. 2/2 (*a*) or (*b*). In Fig. 2/2 (*a*) we have a diode whose anode is connected to the terminal *A*, and whose cathode is connected to one end of a resistance *R*. The other end is connected to the terminal *B*. Now let us connect a battery between *A* and *B*, so that *A* is connected to the positive end of the battery and *B* to the negative end. Then current will flow through the diode and through the resistance *R*, and the circuit will be as in Fig. 2/3 (*a*), i.e. the voltage at the output *C* will be the same as the voltage at *A* and the exact amount of current will depend upon the voltage of the battery and the value of the resistance *R*. *Note*—In all the explanations we shall assume that the diode has no resistance when it is passing current; in practice the resistance of the diode is so much smaller than that of *R* that we can usually neglect it. Now let us reverse the battery so that a negative voltage is applied to *A* and a positive voltage to *B*. The circuit will be as in Fig. 2/3 (*b*). No current at all is passing through the diode, the output terminal *C* is connected to the terminal *B* by way of the resistance *R*, and *A* is disconnected from the output.

The circuit of Fig. 2/2 (*b*) works in exactly the same way as that in Fig. 2/3 (*b*), except that in this case the voltage at *C* will be the same as the voltage at *B* when the positive side of the battery is connected to *A* and the negative side to *B*. When the battery is reversed the circuit will look as though *C* is connected to *A* via the resistance, *B* being disconnected from the output.

Hence we may consider the diode as an ordinary switch which is

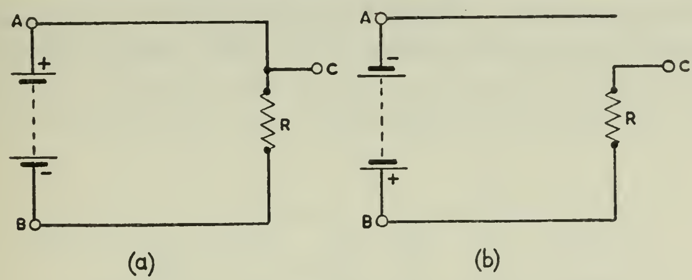


Fig. 2/3. Circuits equivalent to those in Fig. 2/2

closed when the terminal labelled anode is more positive than the terminal labelled cathode, and open when these voltages are reversed in polarity.

It is a simple step from the understanding of the operation of the diode to the understanding of the triode. The only way of controlling the diode current is by reversing the battery and in many cases this is very inconvenient. In a triode, a third electrode called the control

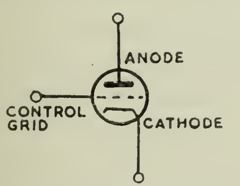


Fig. 2/4. The symbol for a triode

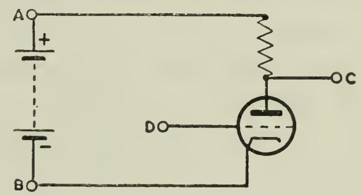


Fig. 2/5. A simple circuit containing a triode

grid is inserted between the anode and the cathode. This is shown schematically in Fig. 2/4.

We have seen how a diode behaves as though it were an infinite resistance (open circuit), or a very low resistance, depending upon the polarity of the voltage applied to the terminals *A* and *B*. The control grid of the triode gives us a means of varying the resistance from cathode to anode between a small and an infinite value merely by altering the voltage applied to this control grid.

Fig. 2/5 looks very similar to Fig. 2/2 (*b*) except for the control terminal *D*, and in fact if *D* were connected to *B* it would operate in a similar fashion. When *D* is at the same voltage as *B* then the valve will pass a large electric current and *C* will be very nearly at the same voltage as *B*. *Note*—The resistance of a triode is considerably greater than that of a diode, and in practice this resistance can never be ignored. If we made *D* about 10 V negative to *B* then the valve cannot pass any current at all and the circuit behaves as though *C* were connected to *A* via the resistance *R*. This transition between no current and full current does not take place abruptly, but there is a gradual change over the range 0 to − 10 V of the control grid *D*.

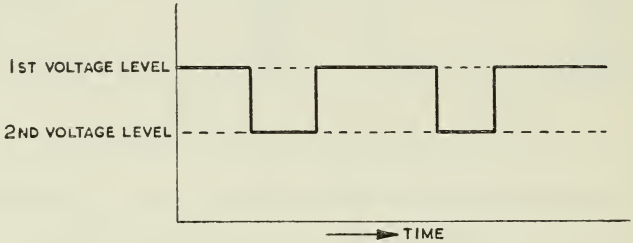


Fig. 2/6. A series of negative-going pulses

If we replace the diode in Fig. 2/2 (*a*) by a triode we shall obtain similar results, except that we must remember that the voltage applied to the control electrode *D* must be measured with reference to the cathode and not to terminal *B*. We may easily reproduce the conditions in Fig. 2/3 (*a*) and (*b*), for if we connect *D* to *C* we shall very nearly have the conditions of Fig. 2/3 (*a*) (remembering that the triode has a certain amount of resistance). Similarly by making *D* 10 V more negative than the voltage at *B* we get the circuit of Fig. 2/3 (*b*). Hence we may get any voltage we like from *C*, from the voltage at *A* down to very nearly the voltage at *B*, by altering the voltage on the control grid.

All the operations so far carried out by these valves could equally well be achieved by the use of ordinary switches and variable resistances, but for one thing—time. Valves can be switched on and off almost instantaneously—there is, for example, no difficulty at all in switching the current off, or on, in less than one millionth of a second. The fastest mechanical switch is a thousand times slower than this.

In an ordinary wireless set, of course, valves are used as amplifiers, but most valves in computers are used as switches, i.e. they are always *on* or *off*, and are controlled by an input voltage.

In digital computers we handle voltage pulses, a pulse being a rapid change from one level to another, a pause at the second level, and then a rapid return to the first level. A graph of the change of voltage level against time for a series of pulses will look like Fig. 2/6.*

Gate Circuits

The first circuit we shall describe is the one for carrying out the logical operation *and*. This circuit has two input terminals and one

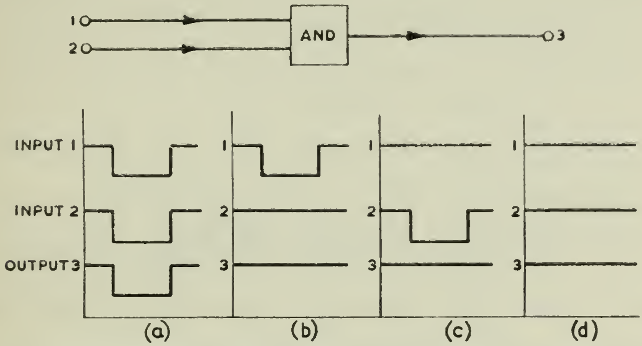


Fig. 2/7. The possible conditions which may be applied to an *And* circuit

output terminal and the condition the circuit must satisfy is that there must be a pulse on the output terminal only when a pulse

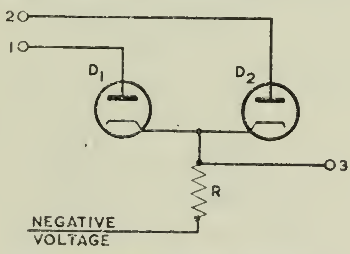


Fig. 2/8. A simple *And* circuit

occurs on both the input terminals at the same time. This is illustrated in Fig. 2/7.

One very simple circuit for obtaining this result consists of two diodes and one resistance, as shown in Fig. 2/8. This circuit will satisfy conditions (*a*) and (*d*) of Fig. 2/7, since these conditions are equivalent to connecting the two input terminals together, and the

* All the pulses which are mentioned are negative-going from earth unless otherwise stated, and about 30 V in amplitude.

resistance of the diode when passing current is very low compared with the resistance R. Under these conditions the two diodes D_1 and D_2 will each pass one half of the current passing through R.

Let us now consider the conditions (*b*) and (*c*) of Fig. 2/7. In the case of 2/7 (*b*), there is a pulse on terminal 1, but no pulse on terminal 2. When the anode of D_1 goes negative with the pulse, the cathode of

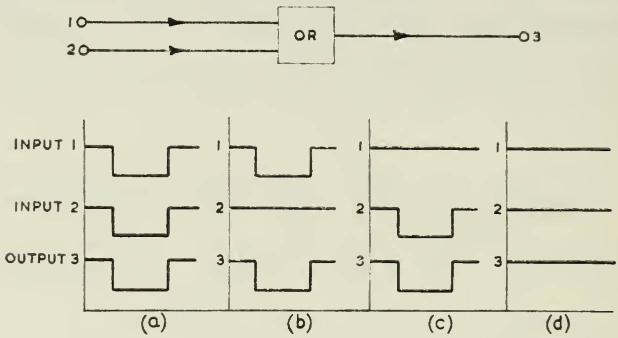


FIG. 2/9. The possible conditions which may be applied to an *Or* circuit

D_1 cannot follow since the current which was flowing through D_1 now flows through D_2, i.e. D_2 now takes all the current flowing through R, and since terminal 2 has remained at a constant voltage, there is no change of voltage at the output terminal 3. The same

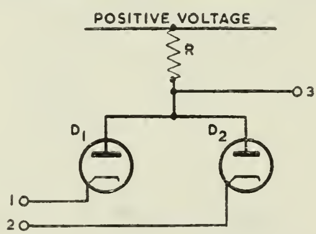


FIG. 2/10. A simple *Or* circuit

conditions apply exactly in the case of Fig. 2/7 (*c*), but with the roles of the diodes reversed.

The second circuit is that used for carrying out the logical operation *or*. The circuit again has two input terminals and one output terminal, and must satisfy the conditions that when there is a pulse on one, or the other, or both the input terminals, then there must be a pulse on the output terminal. This is illustrated in Fig. 2/9. A circuit for satisfying these conditions is shown in Fig. 2/10.

This circuit satisfies the conditions (*a*) and (*d*) of Fig. 2/9, since the two input terminals may be connected together and one of the diodes removed.

Consider the case of Fig. 2/9 (*b*). Here the cathode of D_1 is taken negative. As the resistance of D_1 is small compared with R the anode will also go negative with it. The cathode of D_2 remains at a constant voltage whilst the anode goes negative with the anode of D_1. Hence D_2 ceases to pass current and the voltage on the output terminal 3 is entirely determined by the voltage on terminal 1. Similar conditions apply if terminal 1 is kept at a constant voltage and terminal 2 has a pulse applied to it. Hence this circuit satisfies all the conditions of Fig. 2/9.

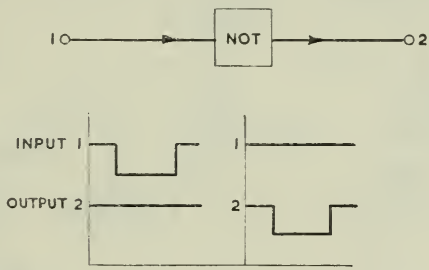


Fig. 2/11. The possible conditions which may be applied to a *Not* circuit

The third circuit carries out the logical operation *not*. There is only one input terminal to this circuit, and there is an output pulse only when there is *not* a pulse on the input terminal. Conversely, there is *not* a pulse on the output terminal when there is a pulse on the input terminal (see Fig. 2/11).

All the circuits used in achieving the *not* operation require a further input which is a standard pulse, which is always present, and like the P pulses is derived from the clock or timing unit which controls the machine. Two possible circuits are shown in Fig. 2/12.

In Fig. 2/12 (*a*) we have two triodes V_1 and V_2 connected with the anode of V_1 to the cathode of V_2, and with the output from the anode of V_2. The standard pulse on the grid of V_2 occurs at the same time as the possible pulse on the input terminal 1, but instead of going negative during this period, goes positive from a negative potential. Hence V_2 is able to pass current only during the period when a pulse may occur in terminal 1. If, however, there is a negative pulse on terminal 1, V_1 cannot pass current and therefore there is no current in the circuit V_1, V_2, R; hence there is no pulse at terminal 2.

If there is no pulse at terminal 1, V_1 is able to pass current all the time, V_2 can pass current during the pulse period, and there will be a negative pulse output at terminal 2.

An alternative circuit is shown in Fig. 2/12 (*b*). This consists of a triode valve and an *and* circuit of the type previously described. The standard pulse has the same polarity and voltage levels as the input pulse, and the resistances R_2 and R_3 ensure that when the valve V_1 is not passing current, the anode of D_1 is slightly positive, and when the valve V_1 is passing current the anode of D_1 is negative, so

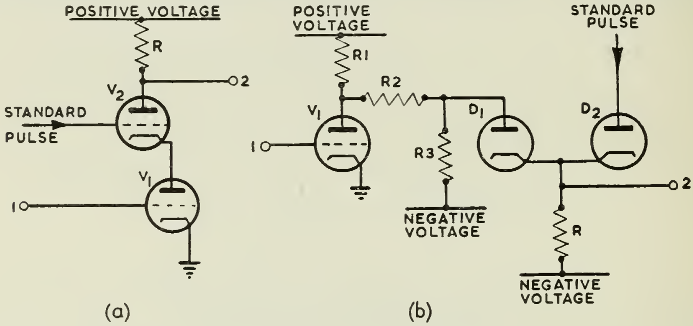


FIG. 2/12. Possible circuits for a *Not* operation

that the standard pulse will pass through D_2 and appear at the output terminal only when the grid of V_1 is positive.

MEMORY CIRCUITS

These circuits "remember" for a considerable time whether or not at a particular instant a pulse was present on the input terminal. One of the simplest will be described first. It may be used in all applications, although it is not always the most economical circuit to use. It has two input terminals, known as *set* and *reset*, and two output terminals, usually called the 0 and the 1 terminals. The reset terminal is used to ensure that the circuit is in a standard condition (the 0 condition) before the presence or absence of a pulse on the set terminal is examined. If there is not a pulse on the set terminal, then the circuit will remain in the 0 condition. If there is a pulse on the set terminal, then the circuit will change over to the 1 condition. The symbol for this circuit and the output voltages for the two conditions are shown in Fig. 2/13.

Before we consider the operation of a flip-flop, as this circuit is usually called, we will describe a simple mechanical analogy which will demonstrate the method of operation. Suppose we have a length of glass tubing, sealed at each end, with a steel ball which is free to roll inside the tube. We pivot this tube at its mid-point over a

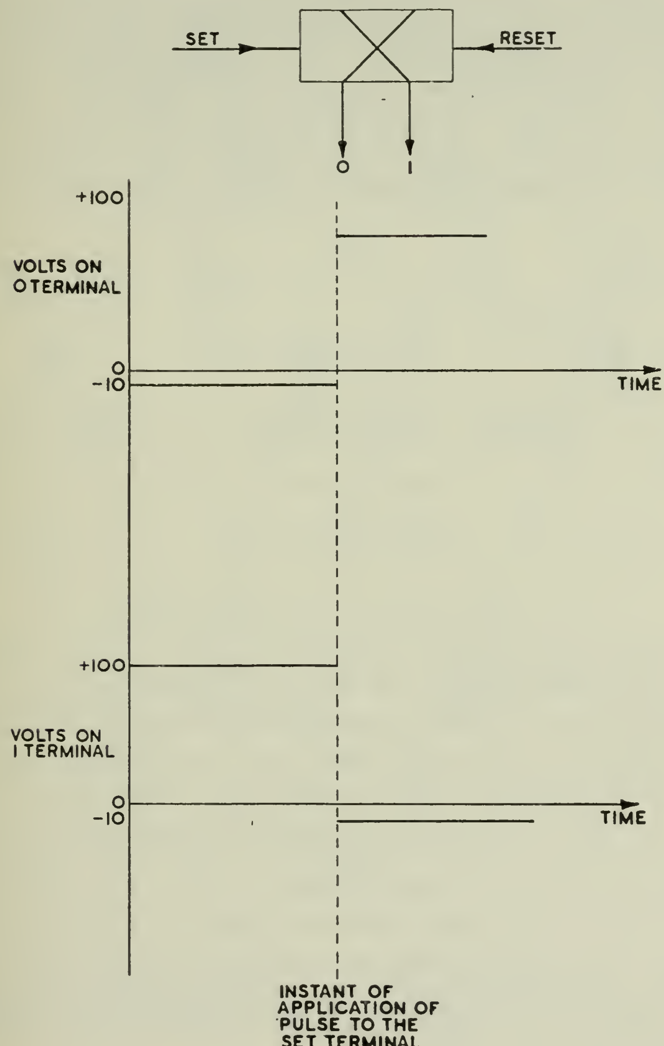


Fig. 2/13

wedge-shaped block, as shown in Fig. 2/14. We will call the right-hand end of the tube the 0 end, and the left-hand end the 1 end. If we position the tube as shown at Fig. 2/14 (*a*), with the steel ball at the lower (0) end, then the tube will be pressing against the wedge-shaped block, and will therefore be quite stable, and remain in this position for as long as we desire.

Now, suppose we start to push downwards on the 1 end, as shown by the arrow *A*. As the 0 end is the heavier end, because of the position of the steel ball, the tube will resist this force and will endeavour to return to its original position. As we continue to push downwards the tube will first become horizontal, and then the 1 end will be lower than the 0 end. As soon as this happens the steel ball will roll from the 0 end to the 1 end, which will then be the heavier end, and the tube will fall into the position shown in Fig. 2/14 (*c*), where it

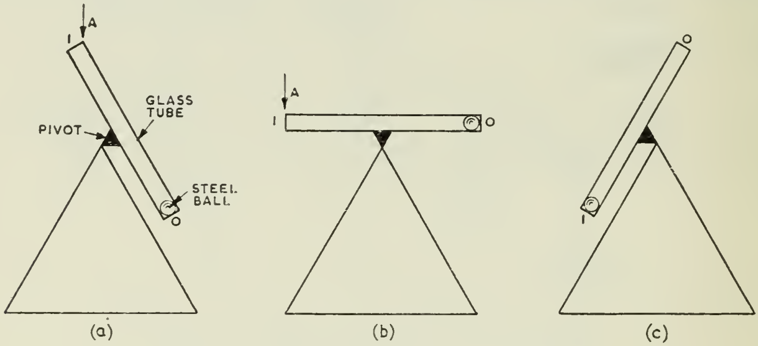


FIG. 2/14. A mechanical analogue of a flip-flop

will remain until some force is applied to the upper or 0 end of the tube. Hence in changing over the state of the apparatus, we move the apparatus part of the way under the influence of an external force, and then the apparatus takes over and completes the change-over without reference to the external force, which may, in fact, be removed as soon as the steel ball has moved past the pivot point.

The electronic circuit equivalent of this model operates very much faster, of course, and instead of requiring a force to be applied, operates under the control of a voltage pulse. In the circuit shown in Fig. 2/15 (*a*) pulses starting at -10 V and rising to 0 V are used for setting and resetting the circuit and are applied at terminals 1 and 2 respectively. Thus the valves V_1 and V_4 pass current only when the pulses are applied.

To describe the operation of the circuit we must assume that it is in one of its two possible states and see what conditions are in operation. We will assume that the grid of V_3 is at the same voltage as the cathode — 0 V, i.e. V_3 is passing current. This means that the voltage on the anode of V_3 is low, due to the current flowing through the resistance R_2. This in turn means that the voltage at the grid of V_2 is negative, because of the resistance chain R_4, R_5. As the grid of V_2 is negative, V_2 is not passing any current, and therefore the anode voltage is high, since there is no current flowing through R_1. Therefore, the grid of V_3 is at o V because of the resistance chain

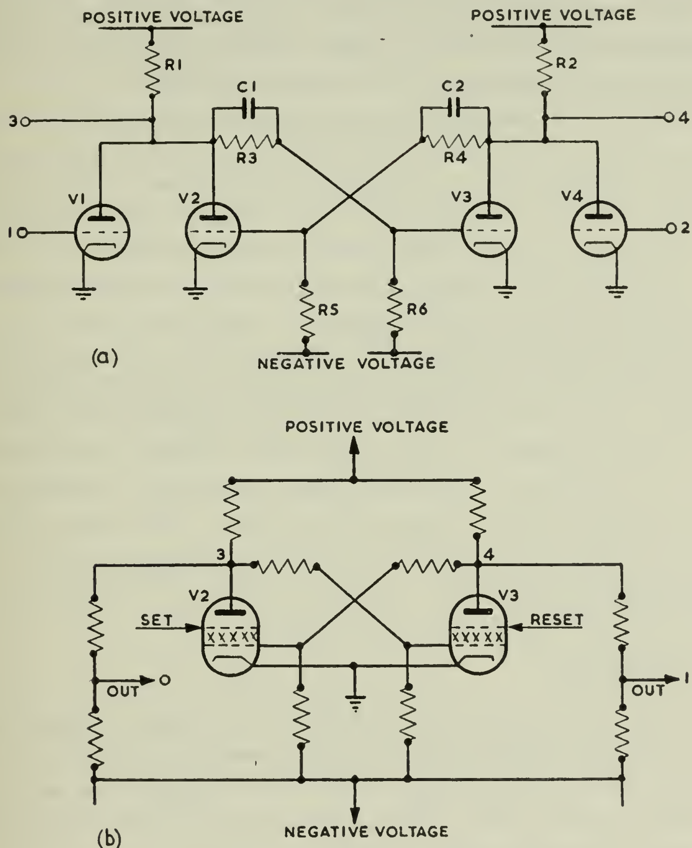


FIG. 2/15. Flip-flop circuits

R_3, R_6, which is the condition we originally assumed. Hence the circuit will remain in this state until we apply some external influence to it.

Suppose we now apply a positive voltage pulse to the grid of V_1. This will enable V_1 to pass current, and because of the consequent voltage drop in R_1 make the grid of V_3 negative. This, in turn, means that the anode voltage of V_3 will rise, removing the negative voltage from the grid of V_2 and therefore allowing V_2 to pass current. At this moment both V_1 and V_2 are sharing the current. If we now switch off V_1, V_2 will take all the current, and the circuit will be stable in its new state, that is, with V_2 passing current instead of V_3.

To reverse the state of the circuit again, we merely require a positive pulse on the terminal 2, the grid of V_4, when the exact reverse of the above operation will occur. The changeover from one state to the other is virtually instantaneous—in practice, times of less than one millionth of a second are easily obtained, and the condensers C_1, C_2 make the operation of the circuit even faster.

The flip-flop as described required positive-going pulses to set and reset it. This is sometimes inconvenient, as the pulses used in a particular computer may be negative-going pulses. In this case another form of the circuit, using valves of the pentode type may be used. For the purpose of explanation, we may describe a pentode as a valve which has two control grids; if either is made negative with respect to the cathode, then no current can flow to the anode. The circuit for a pentode flip-flop will be very similar to the circuit shown in Fig. 2/15 (*a*) but V_1 and V_4 are no longer required, and the input terminals 1 and 4 will now be connected to the second control grids of the valves V_2 and V_3. The operation of this circuit shown in Fig. 2/15 (*b*) is similar to that previously described, except that after a negative-going pulse is applied to terminal 1 the state of the flip-flop will be such that the valve V_2 is not passing current, i.e. the voltage on terminal 3 is high, and the valve V_3 will be passing current, so the voltage on terminal 4 will be low.

The "Staticisors"

It is often necessary for a machine to "remember," and to use for some time, all the information which is contained in a chain of pulses. This can be done by the use of "*staticisors*." Suppose we are working with a chain of five pulses. The appropriate staticisor will consist of five flip-flop valve circuits.

We shall denote the valves which comprise the circuits by A_1, B_1, A_2, B_2, etc. The reader will recall that these flip-flops have two stable positions, and they can be triggered from one position to the other. One valve of each pair takes all the current which is passed by the pair, and the anode potential of each valve is at either one or other of two definite potentials. Suppose that initially the valves A_1, A_2, etc., are passing current. The grids of these valves are all connected to a series of *and* gates each of which has two input terminals. The successive P pulses are fed to these *and* gates, P_1 to the gate of A_1, P_2 to the gate of A_2, etc. The pulse chain which we want to "staticise" is fed in parallel to all the other input terminals of the *and* gates. If the pulse chain contained a pulse in the first interval P_1 the first flip-flop will be triggered; a pulse in P_2 will trigger the second flip-flop, and so on. Suppose for example the pulse chain is 11001. A pulse exists during the first, second and fifth P pulses, so the first, second and fifth flip-flops will be triggered. The flip-flops will then retain, or if we prefer to use the word, they will "remember" the information in the pulse chain until they are reset. It is convenient to say that the "dynamic" pulse chain has now been "staticised."

How can we make use of this information? Two steady potentials are available from each of the five flip-flops. Suppose that, by using a suitable potential divider, we make the potential of an output stage attached to each anode move from + 100 V to − 10 V as the flip-flops change their state. Suppose that we have five equal resistors connected from five cathode followers to the grid of another valve. The valve will pass current, and its grid will not be turned off until all the cathode followers which are connected to it are negative (see Fig. 2/16). Now a grid carrying five resistors, each of which can be soldered to one or the other of the two output cathode followers of a flip-flop can be connected up in $2^5(= 32)$ different ways. This means that it is possible to connect 32 different valves to the cathode followers in such a way that each one can be turned off individually while all the others remain *on*. Each valve can be used to control a single gate, or two or more gates if necessary, so here we have a method by which the course of one chain of pulses can be directed through the machine in a manner prescribed by the information contained in the preceding pulse chain.

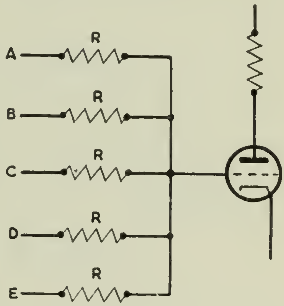


FIG. 2/16. Another *And* gate

This can be used to interpret the output of a staticisor. The potentials of points A, B, C, D, E, are all either + 100 V or − 10 V. The grid is positive unless all five potentials are simultaneously negative

All the circuit elements which have been described so far have worked on the basis of *pulse* and *no pulse*, i.e. there are only two alternatives, or states, and that is why these circuits are used for computation in the binary system.

ARITHMETIC CIRCUITS—THE ELECTRONIC MILL

We now have to examine the rules which must be obeyed if arithmetic operations are to be carried out, and we will now consider the various operations themselves.

Addition

In the addition of two numbers, in any system of notation, the resultant sum of any column depends not only upon the numbers in that column, but also upon the number carried over from the previous, less significant, column. Hence in the case of binary addition with serial pulse trains, we must carry out the addition digit by digit, starting with the least significant, and have a system for remembering the carry digit from the previous digit summation. The rules for addition are easily obtained and may be tabulated as in Table III.

TABLE III

RULES OF ADDITION

x_n	y_n	c_n	$(x+y)_n$	c_{n+1}
0	0	0	0	0
1	0	0		
0	1	0	1	0
0	0	1		
1	1	0		
0	1	1	0	1
1	0	1		
1	1	1	1	1

In this table x_n and y_n are the two nth digits of the serial pulse train; c_n is the digit carried over from the $(n-1)$ position; $(x+y)_n$ is the resultant output, and c_{n+1} is the digit to be carried into the next digit position. As can be seen from Table III the rules of addition are four in number, and may be stated in words as follows—

1. When there is no pulse on any input, there is no output pulse or carry pulse.

2. When there is a pulse on only one of the three inputs there is an output pulse, but no carry pulse.

3. When there is a pulse on two of the three inputs, there is a carry pulse but no output pulse.

4. When there is a pulse on all three inputs there is an output pulse and a carry pulse.

One form of adder using logical elements is shown in Fig. 2/17, and the correctness of operation may be easily tested by applying the above rules to it—

(*a*) No pulse on any terminal. A pulse therefore is emitted by the *not* circuit 7. This pulse will pass through the *or* circuit 8, and be one input to the *and* circuit 9. As neither x nor y nor c_D are present, however, there will be no output from this *and* gate. As there is no pulse present anywhere else in the circuit, rule 1 is therefore satisfied.

(*b*) A pulse on one terminal only. If the pulse is on either x or y, then there can be no output from the *and* circuits 1 and 2. There will be an output from the *or* circuit 3, but as there is no carry pulse

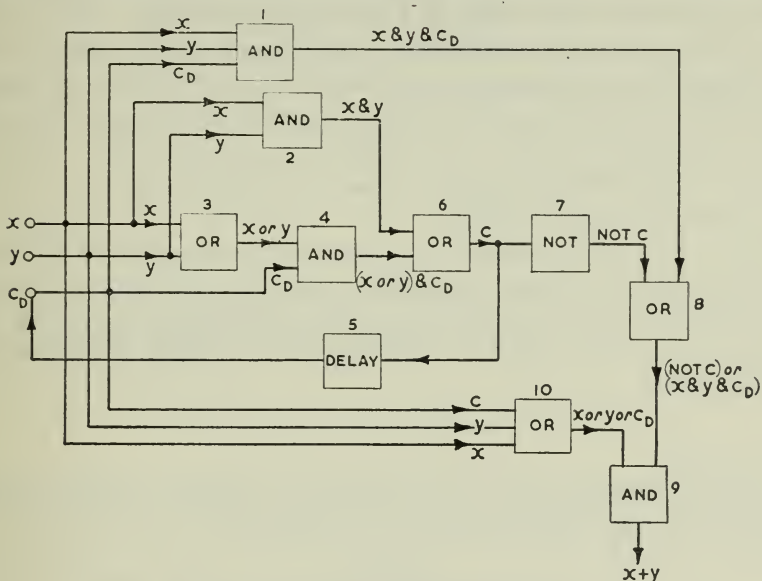


Fig. 2/17. An adder using logical elements

there will be no output from the *and* circuit 4, and therefore c will not be present. There will therefore be a pulse emitted by the *not* circuit 7 and appearing on one terminal of the *and* circuit 9. There is also a pulse on the output from the *or* circuit 10 and therefore on the other input to the *and* circuit 9. Hence there is an output pulse. If the pulse is on c_D there can be no output from 4, as before, and we therefore have the same result—an output pulse with no carry. Hence rule 2 is satisfied.

(*c*) A pulse on two of the terminals. If the pulses are on x and y then there will be an output pulse from *and* circuit 2, and not from *and* circuit 4. There will be an output from *or* circuit 6 and therefore a carry pulse which goes into the delay circuit 5 to generate the carry pulse for the next digit. As there is a pulse on c there is no output from the *not* circuit 7 and hence no output from the *or* circuit 8. Therefore, although there is an output from the *or* circuit 10, there is no output from the *and* circuit 9. If there is a pulse on x or y and on

c_D, there will be no output from *and* circuit 2, but there will be an output from *and* circuit 4, which, via *or* circuit 6, will produce a c pulse as before, the remainder of the operation being identical. Hence the circuit satisfies Rule 3.

(*d*) A pulse on all three terminals. There will be a pulse on both the terminals of the *or* circuit 6 and therefore a c pulse is produced, which forms the carry pulse for the subsequent digit. There is an output pulse from the *and* circuit 1, and therefore from the *or* circuit 8, although there is no pulse output from the *not* circuit 7. As there is also a pulse output from the *or* circuit 10, there is a pulse on both the inputs to the *and* circuit 9 and therefore an output pulse. Hence in this case the circuit generates both an output pulse and a carry pulse and therefore satisfies Rule 4.

It is, of course, possible to devise other arrangements of logical elements which are equally satisfactory and the choice has to be made on detailed circuit requirements, as there usually appears to be little difference in the circuit requirements as shown by schematic arrangements.

Subtraction

Systems for subtraction may easily be devised. In this case the circuit must obey the rules shown in Table IV.

TABLE IV

RULES FOR SUBTRACTION

x_n	y_n	c_n	$(x-y)_n$	c_{n+1}
1	0	0	1	0
0	0	0		
1	1	0	0	0
1	0	1		
0	1	0		
0	0	1	1	1
1	1	1		
0	1	1	0	1

As in the case of addition, an arrangement of logical elements may be devised which will perform the operations required by this table. The design of a suitable system is left to the interested reader.

An alternative method is that known as *complement addition.* If the total number of digits comprising a number is n, the complement of the number y is defined as $2^n - y$. Then the addition of the complement of y to a number is equivalent to the subtraction of y from that number. We may therefore carry out the process of subtraction by forming the complement of a number and adding this complement to the second number. As 2^n has its n least digits equal to zero, the x digit must always be zero. Referring to Table IV, it will be seen that if c_n is a 1, then c_{n+1} is also 1 if x is 0. Further for $x = 0$, $c_n = 1$, when y_n is 1, $(x-y)_n$ is 0, and when y_n is 0, $(x-y)_n$ is 1. Hence if we consider y as the input to the complementer, the output must be zero so long as the input is zero, but as soon as a 1 occurs in

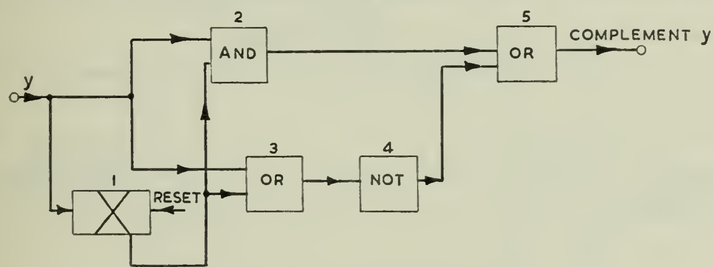


Fig. 2/18. A complementing circuit

the input there must be a changeover in the output which must remain until the end of the number. This changeover causes ones to be emitted for an input of zeros and zeros to be emitted for an input of ones. A simple circuit to carry out this operation may be built up from elements previously described, and is shown in Fig. 2/18.

Before the number is fed into the unit the flip-flop is reset so that its output is negative. This causes the *and* gate 2 to be operated and also gives a negative output from the *or* gate 3. Hence no pulse is emitted by the *not* circuit 4. Zero input will now give zero output until the first pulse appears on the input. This will pass through the *and* gate 2 and the *or* gate 3 and appear on the output terminal. At the end of this pulse the flip-flop 1 is changing over closing the *and* gate 2 and allowing the *not* circuit 4 to come under the control of the input y. Thereafter, until the end of the number, there will be a pulse on the output for no pulse on the input, and no pulse on the output for a pulse on the input. Hence this circuit performs the required operations.

Multiplication

Multiplication in the binary system is extremely easy, as may be seen by Table V. Hence multiplication may be performed by

TABLE V

RULES FOR MULTIPLICATION

		0	1	Multiplicand
Multiplier	0	0	0	Product
	1	0	1	

shifting the multiplicand one digit at a time, which is equivalent to multiplying it by 2, and adding it, or not adding it, into the total,

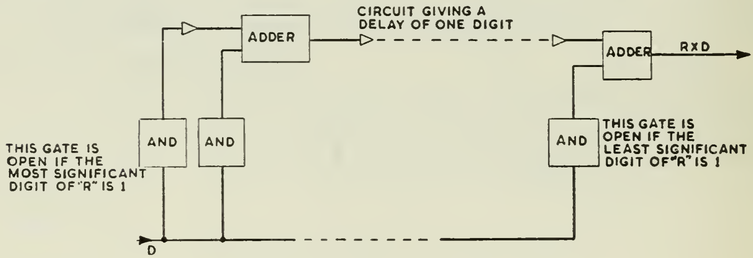


FIG. 2/19. A fast multiplier

depending upon whether the appropriate digit of the multiplier is 1 or 0. A worked example of multiplication is given here—

```
    10101 Multiplicand—D
    10011 Multiplier  —R
    -----
    10101 1 D × 1
   10101  2 D × 1
  00000   4 D × 0 Partial products
 00000
10101     16 D × 1
---------
110001111 R × D
```

Various types of multiplier may be devised, their complexity depending upon the speed with which it is desired to effect multiplication. One of the simplest to understand, which is also one of the fastest, is shown in Fig. 2/19. If there are n digits in the largest numbers to be multiplied, then this arrangement requires n *and* circuits, $(n - 1)$ adders and $(n - 1)$ circuits each giving a delay of one digit. Multiplication takes place in two stages. First of all the multiplier is fed to n flip-flops, each of which controls the operation of one of the *and* circuits. The multiplicand is then fed in along the line marked D in Fig. 2/19, as a serial pulse train. According to whether the *and* circuits allow the pulse train to pass or not, so will the partial products formed consist of either zeros or duplicates of D. These are added simultaneously in the chain of adders, with the delay circuits giving the appropriate shift to each partial product—from the diagram it will be obvious that if D passes through the *and* circuit controlled by the most significant digit of R, it will be delayed or shifted by $(n - 1)$ digits, since there are $(n - 1)$ delay circuits between its input and the output from the multiplier; on the other hand if D passes through the *and* circuit controlled by the least significant digit of R, then there is no shift or delay.

In the type of multiplier described, a delay circuit has been mentioned, although not previously explained. An explanation follows.*

The input waveform is differentiated by R_1, C_1, and the positive-going edge drives the grid of V_1 to 0 V. This grid voltage is held by the condensers C_2 until the next dot waveform, whose leading edge is coincident with the leading edge of the serial pulse or dash waveform. Hence on the anode of V_1 there is a short negative-going pulse in the space immediately following an input pulse. This pulse is again differentiated by C_3, R_3, and the positive-going edge, through the diode D_3 drives the grid of V_2 to 0 V. The grid remains at this voltage until the end of the digit period, when the inverse dash waveform, via the diode D_4, carries the grid negative again. Hence there appears at the anode of V_4 a pulse delayed on the input pulse by a single digit period.

THE MEMORY OR STORE OF THE MACHINE

The efficiency of a machine as a computer will depend very much on the amount of information that can be stored in it. A machine with a poor "memory" can tackle only a very limited range of problems. Babbage proposed to store the equivalent of 150,000 binary digits in his analytical engine, and much of the research which has been done in the development of modern computing machines has been directed towards the production of economical memories capable of storing large quantities of data in the most efficient possible manner. So far, we have considered information in the form of trains of pulses moving through the machine with the speed of light. How can they be stored?

Most modern machines make use of one or other of four types of memory which will be described in turn.

* See Fig. 2/20.

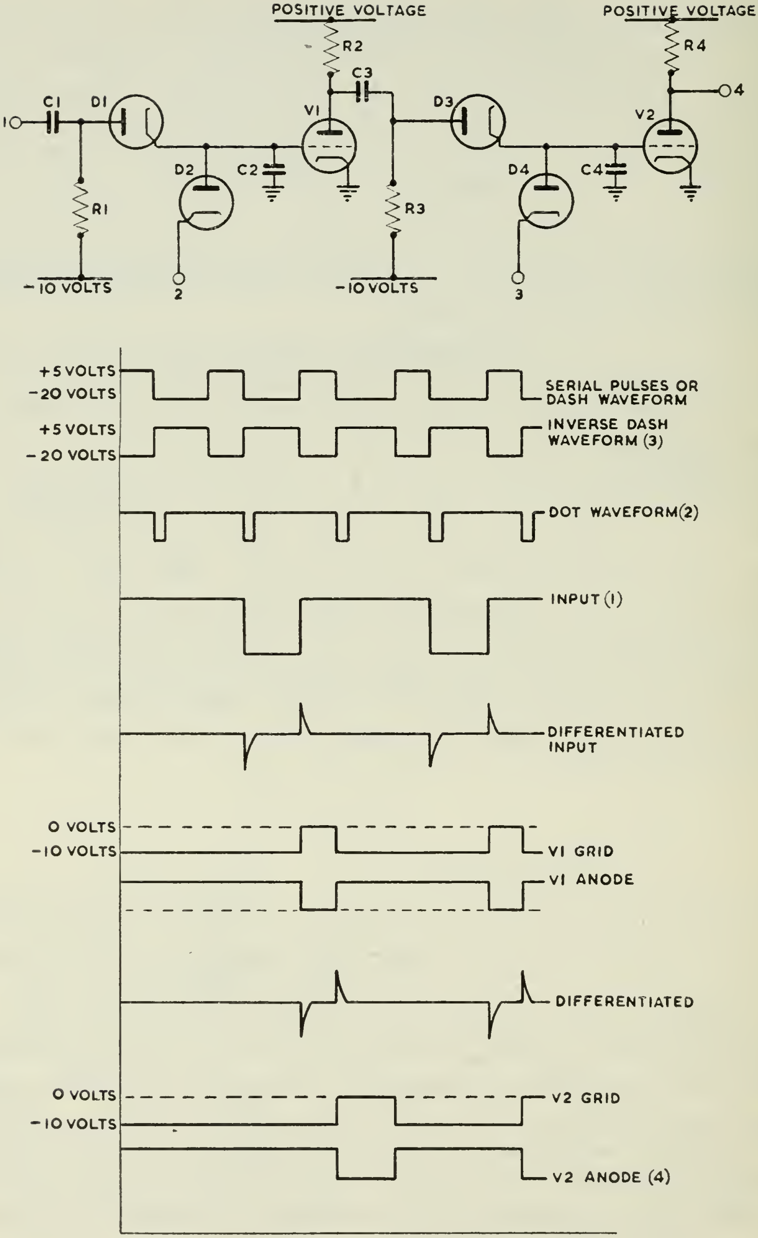


FIG. 2/20. A delay circuit and some waveforms

Staticisor

This has already been described (see page 50), for it is a most valuable circuit element. It would be possible to use staticisors as the main memory of a complete machine, but it will be clear that to store a single digit in a staticisor one needs at least two valves. We need to store thousands if not hundreds of thousands of digits, so it is necessary to devise a less extravagant circuit.

Delay-line Memory

A tube filled with mercury has a quartz crystal mounted at each end. If electric impulses are applied to one crystal it is set into vibration and the waves which it produces in the mercury travel down the tube at a speed of about a mile a second. When the waves arrive at the far end of the tube they fall on the other crystal, set it into vibration and cause a small voltage to be developed across it. If we send a train of electrical impulses into one crystal they will emerge from the other crystal about a millisecond later if the mercury column is five feet long. The emerging impulses can be amplified by valves and fed back into the tube so that they go round and round. In practice if this is done the pulses become distorted after several passages round the tube, but on the other hand it is possible to make each emerging pulse open a gate. A new and undistorted pulse derived from the quartz-oscillator "clock" will pass through the gate and can be applied to the input quartz crystal. The chain of pulses will now pass round and round indefinitely as long as the set is kept *on*. It is, of course, necessary to relate the time of transit of a pulse through the tube to the frequency of the master oscillator which opens the gate. The velocity of propagation of waves in mercury depends on the temperature, and it is usual either to control the temperature closely or to adjust the frequency of the oscillator to suit the temperature of the tube. Elaborate circuits have been devised to do this automatically.[(3)] In one existing machine (the E.D.S.A.C. in Cambridge—see page 132) the pulses last about a microsecond and are a microsecond apart so that approximately 500 pulses are stored in one mercury tank about five feet long. The A.C.E., which is now being built in the National Physical Laboratory (see page 135), uses a similar memory and so does a machine built by the Bureau of Standards in Washington. Whenever we want to read a number which is circulating round the tank we can do so by opening a gate at the moment when the pulse train in which we are interested is about to emerge.

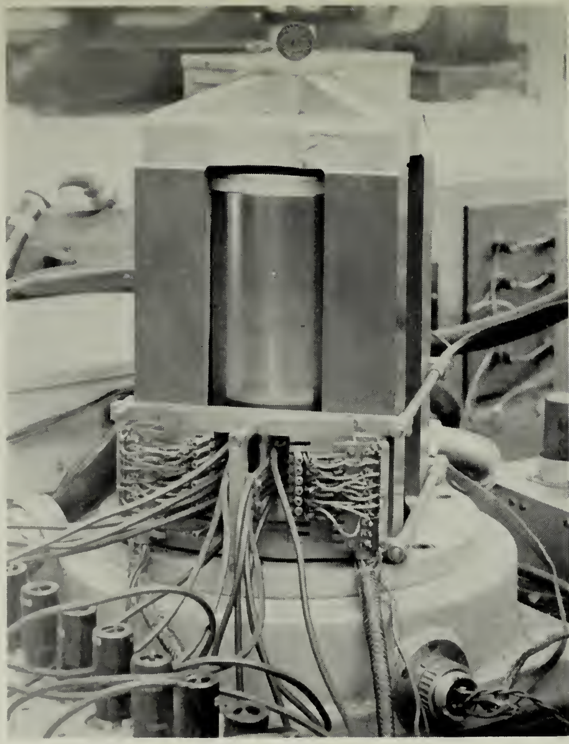
By using a dozen such tanks, we can increase the total storage capacity of the machine correspondingly. Such a memory can be likened to a dozen files; each file can be selected instantaneously, but the contents of a file can be studied only by going through it page by page. The system has been compared to a juggler, who can keep a large number of balls in the air by throwing them up in turn, but who gets hold of any one particular ball which he may happen to want only when it falls into his hand. In brief the important limitation of this type of memory is that it is usually necessary to wait for some time before the pulse train which one wants is actually emerging from the mercury tank (see page 111 for an analysis of the implications of this fact in practice).

Some modern theories of human memory suggest that it may involve processes not unlike those which we have just described—a series of stimuli may pass round and round in a chain, perhaps along several such chains in parallel. The process of learning apparently involves the establishment of closed paths along which pulses of nervous energy may pass, as well as the initial formation of chains of pulses. With advancing years old people find it more and more difficult to wear a new memory path in their minds. A new idea may remain with them for only a few hours, whereas memories of their childhood may abide with them as long as they live. A digital computer with a mercury delay-line memory forgets everything as soon as the power is switched off. An old man may forget when he falls asleep.(4)

The High-speed Electrostatic Store

This unit is the heart of the new Manchester computer, and its properties have determined the layout and the operation of the whole machine. It is being used in the T.R.E. computer (see page 144) and in many computers which are being built in America, most notably in the S.E.A.C. (the Bureau of Standards machine in Washington which is unique in using a delay-line store as well), the S.W.A.C. (the Bureau of Standards machine in Los Angeles) and in the computer of the Institute for Advanced Studies in Princeton. Professor F. C. Williams and Dr. T. Kilburn, who invented this type of store, described it in the *Proc.I.E.E.* (March, 1949). In the interest of simplicity the present account omits all mention of complicating secondary effects, and the original paper(5) must be referred to for more comprehensive information.

Other electrostatic storage systems have been described but this



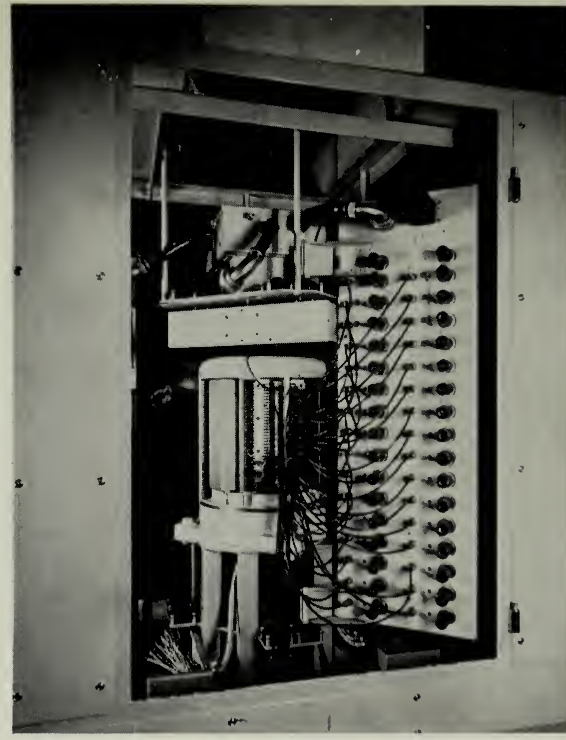


PLATE IV. THE MAGNETIC DRUM OF THE MANCHESTER MACHINE

Left. Before installation. *Right.* After installation

system is unique in that it does not use special storage tubes, but makes use of an ordinary cathode ray tube, which has a pick-up plate mounted outside it, immediately in front of the fluorescent screen.

Suppose we bombard a spot on the screen with a beam of electrons which has been accelerated by a potential between 1,000 and 2,000 V. Under these conditions the coefficient of secondary electron emission will be greater than one. This means that more electrons will be emitted by the spot which is being bombarded than fall upon it from the primary beam, and so the spot will become positively charged. The secondary electrons which are emitted have relatively small velocities—some will fall back from whence they came, some will fall on adjacent parts of the screen, but the majority will be attracted to the conducting coating on the inside bulb, which is usually connected to the final anode of the electron gun.

There is a small capacity between the spot on the screen and the collecting plate so that the result of bombarding an uncharged spot will be to produce a small positive pulse on the collector plate, which can, of course, be amplified to any size required. When we bombard the spot for a second time, however, it will still be at equilibrium potential, so that no positive pulse will be induced in the pick-up plate. We can therefore tell, by bombarding a spot on the tube, if it has recently been bombarded or not. It is possible to write information on to a series of spots on the surface of the tube and subsequently to read it out again. So far, so good, but it will be clear from this argument that in the act of reading from the fluorescent screen we have to write all over it, and erase everything which was written there. Moreover, the insulation of a fluorescent screen is such that any charge distribution which has been built up on it will decay in a few seconds. A cathode ray tube used in this way is not of much use as a memory—let us see what can be done to improve it.

Suppose that immediately after bombarding one spot, A, we move the beam by a small distance, about equal to a spot diameter, and bombard an adjacent spot, B. Slow secondary electrons will be emitted by B, some of them will be attracted to A, which is the most positively charged point nearby, and after a short time they will remove the positive charge on A completely. This means that if we now bombard A for a second time a positive signal will appear in the amplifier. On the other hand if the spot B had not been bombarded since the first bombardment of A, A would still be at its equilibrium potential, and no positive signal would be obtained by bombarding it.

Suppose now that we arrange a mosaic of pairs of spots, which we will call A_1B_1, A_2B_2, A_3B_3, etc. The individual spots in a pair are to be very close together, but adjacent pairs are far enough apart to ensure that secondary electrons from one pair do not affect any other pair on the screen. When we bombard A_1 we get a signal which tells us whether, during the last cycle, B_1 was bombarded or not; moreover this signal occurs a short time before we are ready to bombard B_1 for a second time. We can, therefore, avail ourselves of the opportunity and operate an electronic circuit which turns on the electron beam before the time comes to change the voltage on the deflector plates and direct the beam to B_1. We can, in fact arrange to bombard B_1 if it was bombarded during the last cycle, and to refrain if we refrained last time. By doing this we renew the charge pattern stored on A_1B_1, and in the process we read off what was stored there. If we repeat this operation once or twice a second, the charge which is lost by natural leakage from the screen will be replenished, and the charge pattern will remain unaltered for as long as we please.

The great advantage of this type of "memory" is that, by suitably controlling the potentials on the deflector plates of the cathode ray tube, it is possible to direct the beam almost instantaneously to any part of the screen in which we happen to be interested. Here, it seems, we have a "memory," the contents of which are immediately available to the machine on demand. There are two complications which we must mention at this point. In the first place it is found better not to use the simple arrangement of pairs of dots which has been described. Instead the beam is defocused to cover an area about four times its normal size in order to "read" (this corresponds to A_1) and it is then focused sharply to "write" (corresponding to B_1). The principle of operation is exactly the same as before, but the new A is an annulus surrounding B, and the output signal is three times as big as it is when we use the dot method. The charge pattern will leak slowly away unless it is renewed once or twice a second and the operation of the machine must ensure the regeneration of all parts of the store whether or not they happen to be needed in the computation.

The details of the memory units which are now in use vary from one machine to another, so for convenience we shall describe that which has been built for the Manchester machine. The face of a single cathode ray tube is arranged to hold 64 lines each of which contains 20 pairs of dots (or 20 defocus-focus spots). Each pair stores one binary digit, so that a single tube contains 1,280 binary digits in all. It has been necessary to develop a special cathode ray tube which has an unusually good electron gun. The manufacturers have to take special precautions in order to avoid small specks of impurity which might make it impossible to store digits on some parts of the screen, but the defocus-focus method has the advantage that, since we irradiate a larger area of screen, the effect of small specks of impurity in the screen is relatively unimportant. This very much eases the problems of the manufacturers, and has proved to be of considerable importance to them.

The electron beam is made to trace out the lines one at a time by the potentials which are applied to the *X*-plates of the cathode ray tube, and the choice of the line which is to be scanned is made by adjustment of the potential of the *Y*-plates. In order to ensure the regular regeneration of the whole store, the tube face is scanned in the following manner. Successive 240 μsec intervals are devoted alternately to the regeneration of the store and to reading off the numbers which are needed in the computation. In other words, in alternate *X*-sweeps of 240 μsec periods the whole face of each tube is scanned regularly, line by line, and in the interleaved *X*-sweeps, the machine refers to that line on the particular storage tube from among those in use on which is stored the information which is needed next. Thus, half the time is devoted to regeneration of stores and half to computation. The time which has to elapse between the decision by the machine that it needs a piece of information and the instant at which it begins to read it out is very short indeed, and it is to this fact that the speed of the machine is primarily due.

Another method of exploiting this type of store is to be found in the machine which is now being built in Malvern (see page 144). It may be worth mentioning a third variant in use in Washington. In this computer, 45 Williams memory tubes are connected in parallel. The tubes each carry 256 spots. By appropriate adjustment of *X* and *Y* deflector potentials the information on one of the 256 spots can be read simultaneously from each tube. It is made to set up a series of 45 flip-flops, which are then inspected in turn by the *P* pulses, so that the information in the memory tubes, which has been stored in parallel, is converted into a pulse chain which can be handled in the usual way. Writing into the tubes is done by reversing the process, and the operation provides for alternate cycles of regeneration and computation.

The Magnetic Store

So far we have concerned ourselves with the memory which contains information required to be immediately available. If we use the analogy of the human computer, this corresponds to the book of tables which he has on his desk, to his working instructions (which may be in his head) and to the paper on which he records the results of intermediate calculations. In addition to this the computer needs occasional reference to a main library, which contains, for example, formulae or functions not needed frequently, instructions which enable him to perform certain standard computations, or tables of empirical data and experimental results. It is in this part of the machine that the requirement for a large store capacity is vital and it is usually uneconomic to use any of the "memories" which we have discussed so far to provide it.

Several machines have "memories" based on principles which have been applied in recording sound on magnetic wires and tapes. The design which has been most frequently used stores information not on wire but on the surface of a metal drum coated either with metal or with a layer of iron oxide. In order to be explicit we shall describe the drum which is being used in Manchester, as this is quite typical. The drum is about ten inches in diameter and twelve inches high, and is coated with a layer of nickel 0·00025 in. thick. Two hundred and fifty six writing and reading heads are mounted one above the other in eight rows, each parallel to the axis of the cylinder, and the nickel surface moves past the heads at a speed of 60 m.p.h. (approximately) and is about 0·0005 in. away from them. This high speed makes it possible to record at 100 kc/s, which is the normal speed of operation of the Manchester machine. When it is necessary to "write" on to the drum, one of the writing heads is chosen (by means of a series of relays) and then, at a time which is determined by the quartz master oscillator, a current is passed through the head, first in one direction and then in the other. The digits 0 and 1 are distinguished by the order in which the currents are sent through the head. Digits exist in the drum in the form of little magnets, and the magnetic system, once it has been set up, will last indefinitely. It is important to stress this fact, as the other memories which we have described lose their contents once the machine is switched off, or the power fails.

As the magnets pass beneath the reading heads they induce very small voltages in them. The voltages can be amplified, and by observing which direction of magnetization comes first in a digit period, the machine reads out 0 or 1. The operation of observing a waveform at a particular instant of time, which is controlled by some other waveform derived for example from a clock, is known as *strobing* the original waveform. It is a process which has become familiar through its use in radar circuits, and is frequently used in digital computers.

Each reading head has a separate pre-amplifier which is driven by a step-up transformer. The choice of the track from which one wishes to read is made by switching on the appropriate pre-amplifier (this can be done almost instantaneously). It is then necessary to wait for a time which may be as long as 30 ms (which is one revolution of the drum) before the required information becomes available. In order to avoid the time which would be wasted by frequent access to the drum, it is normal to transfer information in large blocks to the high-speed memory before use.

It will be clear from the account which has been given of the operation of the magnetic memory that it is necessary to keep it in synchronism with the main machine, so that a signal which has been written will emerge during the corresponding clock pulse pattern when it comes to be read. This can be achieved in two ways. It is possible to use the drum as a source of the waveforms which control the whole machine—this is probably the most common method of operation; alternatively the speed of the drum can be controlled by a servo system which keeps it in the proper phase relation with the *P*-pulses which we have mentioned before. This has been done in the Manchester machine.(6) The maximum error in the position of the drum at any specified instant must not exceed about a tenth of the length occupied by a single digit, that is to say about a thirty-thousandth of the circumference of the wheel, which means that it must be in a specified position within 1 μsec of the correct instant.

The number of digits which can be stored on a drum is roughly proportional to its area divided by the manufacturing tolerances that can be maintained in building it. In order to pack 3,000 digits into a square inch of magnetic material 0·00025 in. thick, it has been found necessary to balance the drum itself, which weighs 100 pounds, to better than one part in 200,000, and to machine it to a maximum eccentricity of less than 0·0001 in. The problems which arise in building a drum are very similar to those which arise in the construction of large gyro compasses.

The illustrations, Plates IV, VIII, IX and X, are of the Manchester drum.

REFERENCES

1. CHERRY. *Proc.I.E.E.* **98,** Pt. III, 55 (Sept., 1951).
2. CARRINGTON, J. F. *The Drum Language of the Lobele Tribe* (1944).
3. PINKERTON, J. M. M. *Electronic Engineering* (Apr., 1951).
4. YOUNG, J. Z. *Doubt and Certainty in Science* (1951).
5. WILLIAMS, F. C. and KILBURN, T. *Proc.I.E.E.* **96,** Pt. III, 40 (Mar., 1949).
6. WILLIAMS, F. C. and WEST, J. C. *Proc.I.E.E.* **98,** Pt. II (Feb., 1951).

Chapter 3

THE ORGANIZATION OF A TYPICAL MACHINE

A computing machine in action is more than the concatenation of relays and storage mechanisms which the designer has built into it—
Norbert Wiener

The underlying principles of all computers are very similar, but it will be necessary in this chapter to use one particular machine to illustrate these principles, otherwise our explanations would be so general as to be virtually meaningless. For this reason we shall refer most frequently to the new Manchester University machine, which is representative of the best modern practice.*

Let us see how we could build up a machine from the components which we discussed in the last chapter. First of all we must remind ourselves that all information travels through the machine in the form of chains of pulses. The Manchester machine contains cathode-ray-tube stores in which a pulse chain can be "remembered" until it is wanted. One such chain can be passed into a staticisor, "remembered" and used to control gates which define the path of the next pulse chain—recalling the analogy of an automatic telephone exchange.

Suppose that we had four memory tubes, each of which had thirty-two lines on it. We know that the anode potentials from a two-pair staticisor will open one of the four gates which allow us to select the output from the tubes. In a rather similar way the output potentials from a five-pair staticisor set up a potential on the *Y*-plates

* We apologize for the repetition of much of the subject matter of this chapter elsewhere in this book; it has been our experience that the layman finds it very hard to grasp and to follow an account of the operation of a computer, and that he finds it helpful if the whole subject is presented to him several times, particularly if successive treatments are more and more sophisticated. This is the method which we have tried to follow, and earlier chapters have been ruthlessly simplified as far as possible. We beg, therefore, if the repetitions are found excessive and irksome, for indulgence as our pedagogical methods will have failed.

The reader is assured that a certain amount of repetition has been deliberately introduced, and he is recommended, if any part of the account is found particularly obscure, to omit it entirely. An explanation may follow later in the book. In any event it is quite unnecessary to follow all the details of circuits and things; if the fact can be appreciated that circuits exist, and can readily be built, which will perform certain specified functions, that is all that is necessary in order to follow the rest of the book.

of the tubes, which makes the beam scan one particular chosen line on each tube. Any one of the 128 lines on the four tubes can therefore be picked out by a proper combination of the two staticisors, and we recall that these staticisors will have been set up by the pulse chain which has just passed into them.

Another staticisor can be made to open the chosen gate into the mill, or arithmetic unit, which contains an adder, a multiplier, a collator and other units of the type described in the last chapter. The number selected from the store is passed into the mill, and after some operation has been performed on it, the result can be put back into the store. It is customary to put the result of each arithmetic operation first of all in a special line in the store which, by analogy with an ordinary desk calculator, is called the *accumulator.* The number in the accumulator can be put back into the store in a separate operation.

If two numbers N and M are to be added, for example, one of them (N) is written into the accumulator in a first operation; in a second operation the chain of pulses which represents N (from the accumulator) is presented together with the chain of pulses which represents M (a chain which will be coming from a place in the store selected by the control circuits in the machine) to the two inputs of an adder (see page 53). The chain of pulses coming *from* the adder, which represents the sum $N + M$, is written into the accumulator, digit by digit, into the very same place from which N is emerging, digit by digit. The adding circuit works quickly enough to find the sum of the digits which are presented to it, during the interval between the "reading" and the "writing" pulses which are involved in the production of one dot-pair in the store (see page 60).

The machine is able to devote a short time to making up its mind about the sum of the digits which are presented to it, e.g. 0 and *carry one.* This period is conventionally called the *meditation time* of the machine, and the longer it is the better. Even the most recalcitrant adding circuit can make up its mind quite definitely in the microsecond or so which is available in the Manchester machine between the "read" and "write" waveforms.

One cathode-ray-tube store is used to *control* the machine. On one line is stored the serial number of the instruction which the machine is obeying. During the regeneration cycle of this tube, which takes place after the machine has just completed an operation, the number 1 is added to the contents of the control tube so that the machine will usually obey a number of instructions in sequence.

The pulse chain which emerges from the control tube is used to set up staticisors which select a particular time in the main store. This line, containing the instruction which must next be obeyed, is then copied into a special line called the *present-instruction line* which, for convenience, has been put on the control tube. The present instruction is then read out again and sets up all the necessary staticisors, which open the appropriate gates, choose the number on which the operation is to be performed and route it to the correct destination.

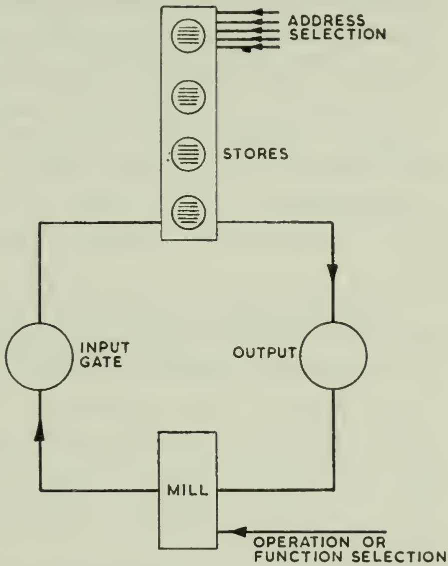


Fig. 3/1. Schematic diagram of a typical machine

Thus there are four operations in all, and these may be summarized as follows—

1. Read the number in the control tube, add unity, and use the resultant pulse chain to set up staticisors which select the next instruction to be obeyed.

2. Copy the instruction into the present-instruction line.

3. Re-read this information, and use it to set up the staticisors which choose the arithmetic function and the number on which it is to be performed.

4. Read out the number on the chosen line, and pass it through the appropriate part of the machine.

It may appear from this account that several sets of staticisors have been used in the operation, but this impression is misleading. There is, in fact, one set each of *function* staticisors, of *tube* staticisors, and of *line* staticisors, and each set is used exclusively to perform the operation which its name suggests. It is in order to avoid the duplication of staticisors that it is necessary to copy out the information from the main store which contains it into a subsidiary store (i.e. the present-instruction line) before it is used.

Analysis of the operations which have just been outlined will show that in operations 2 and 4 information is being read into or from specially chosen lines. In operations 1 and 3 the main stores are not required by the machine. It is possible, therefore, to make use of the time to regenerate the main stores in sequence, line by line.

It is customary to speak of the cycle of operations as a *four-beat bar.* Successive beats are called *Scan* (1), *Action* (1), *Scan* (2), and *Action* (2). Actual numerical computation takes place only during the last beat of a bar which accordingly may have to be made of variable length.

All the pulse chains with which the machine deals are fundamentally similar, but the machine interprets some as instructions, and some as numbers, at the will of the programmer, who determines the part of the timing cycle during which they are used.

So far we have considered a machine which will obey instructions in regular sequence. Such a machine would be limited in its speed by the rate at which a programmer could punch the tape containing its instructions. This process is so slow that most people would prefer to use a desk calculator. How can the machine be speeded up?

In the first place it is possible to make it perform one sequence of operations several times in succession on different numbers if we (*a*) change the number in the control tube back again to that corresponding to the beginning of the sequence in question, and (*b*) give the machine a new set of numbers to work with. The first of these two operations can be performed by writing into the control tube the number required. After all, the control tube is very like an ordinary store, so we can write into it any number we please.

Babbage devised a scheme by which any batch of cards could be used automatically by his engine as often as it was needed to repeat the series of instructions (or subroutine) which they embodied, and Lady Lovelace suggested that: "In preparing for calculations by the engine it is desirable to arrange the order and combination of the processes in cycles in order that its mechanical advantages may be exploited to the utmost . . . the power of repeating the cards reduces to an enormous extent the total number of cards required."

The second requirement can be met if, without otherwise modifying the programme, we can alter the addresses of the numbers on which the machine is to operate. It is so often necessary to make this change in a list of instructions that in almost all machines a standard procedure has been devised to do it. The simplest method to describe is probably that used in the Manchester machine, which includes a special piece of equipment known as the *B-box.* A series of numbers is stored in this *B-tube* and can be added to those numbers in the instructions which specify the addresses; an example may make the point clearer. Suppose we want to form the sum of the sines of a hundred angles, which have been stored in the machine in consecutive storage locations (the machine may, of course, have computed these angles itself). We devise a series of instructions for working out sin θ from θ. The machine follows this routine and evaluates sin θ_1, but we do not have to instruct it in detail how to evaluate sin θ_2. We use the same routine as before, but alter the address in the routine which tells the machine where to find θ from that in which we have stored θ_1 to that in which we have stored θ_2. Suppose that the angles have been stored in locations numbered 200 to 299. We have a routine which, as it is actually written, instructs the machine to find the sine of the angle stored in location "zero." We write the number 200 into a line on the *B*-tube, whence the machine automatically adds it to the "zero," and so it finds the sine of θ_1. All that we now have to do is to add 1 to the *B*-line, and make 200 into 201, and the machine will find sin θ_2.

This procedure enables us to find the sum of the sines of our 100 angles with one routine for sines and one simple *B* instruction, and it clearly represents a major simplification in the problems of the programmer.

Our account of the machine is still incomplete. A machine built in the manner we have described would be able to perform a large number of routine calculations at great speed: it would be able to work out the trajectory of a shell if the routine could be prescribed in complete detail. The programmer could do this if he had previously worked out a similar computation by hand, and knew, for example, the number of terms he must evaluate in each of his series, and the size of the numbers with which he was dealing, so that he could take steps to ensure that the numbers he used never outran the capacity of the machine. We have not, however, endowed the machine with any power of independent judgment, so that if the programmer cannot foresee at every stage exactly what form the computation will take, the machine cannot help him.

A hundred years ago, Lady Lovelace wrote of Babbage's machine: "The Engine is capable, under certain circumstances, of feeling about to discover which of two or more possible contingencies has occurred, and of shaping its future course of action accordingly." It is, at first sight, astonishing to find that the machine can be made in this way to guide itself through the complexities of any computation, however elaborate it may be. It does so in effect by using a mathematical version of the game of *Twenty Questions*.

If the reader still has any doubts of the value of the binary system he may recall that there can be very few objects in Heaven or earth that a skilful inquisitor cannot identify by obtaining the answers "yes" or "no" to a few well-chosen questions. The machine can pose only one type of question, but there is no need to limit it to a total of twenty. At certain appropriate points in the calculation the machine examines the content of the accumulator, and observes whether it is positive or negative (or perhaps zero). The programmer will of course have to instruct the machine to put a special number into the accumulator before the test is made; this number may have arisen naturally in the course of the main computation or it may have been specially calculated by a routine devised for the purpose. The alternative courses of action must, moreover, have been prescribed in detail by the programmer, but the choice between them is made automatically by the machine itself in the light of the result which the computation has yielded so far. Here at last is the essential step in the programme; here is the key to the speed and universality of the machine. This is how the machine is able to drive itself forward, as Babbage put it, "by biting its own tail."

Let us consider an example which was used by Babbage to illustrate this very point in 1837. His problem was to find the smallest integer whose square ends in 269696. Babbage thought that the answer he wanted was 99736. It is of course quite unnecessary for the programmer to have this information, and he might instruct the machine to proceed as follows—

1. To find the squares of the natural numbers in order. This is a straightforward process; the machine computes n^2 from $(n - 1)^2$ by adding $(2n - 1)$ to it, and follows the same standard routine time after time.

2. To subtract the squares in turn from 269696.

3. To see if the last six digits of the number so formed are zero. The machine now has to decide which of two possible courses of action it is to follow: (*a*) if the number is not zero the machine must compute the square of the next natural number and try again, and (*b*) if the number is zero, the machine must stop calculating and print out the number whose square satisfies the condition which has been specified.

If the programmer were to use this primitive routine (and it is clear that there are many possible short cuts) the total number of instructions required would be less than a dozen, apart from those necessary for input and output. The machine would go through the routine 99736 times. It might take a quarter of an hour to prepare the routine, punch it, and feed it into the machine; after that had been done, the machine would take about another quarter of an hour to perform the necessary 99736 cycles of operations. This is probably much better than an abacus, but before we congratulate ourselves too warmly it is worth remarking that when this problem was propounded to Mr. Klein, who solved it entirely in his head (see page 312), it took him exactly three minutes to show that the correct answer is 25264, and that Babbage had been wrong. This example, which we included because of its historical interest, is perhaps scarcely representative so we will give three more.

Suppose we are computing some function expressed as a power series. We shall need to construct for the machine a list of instructions which can be divided into two main groups. One of these will result in the formation of one term of the series, and the addition of this term to the previous partial sum. The other group will be necessary for telling the machine when to stop, i.e. how many terms to include in the series.

Consider for example a sine series or a geometrical progression. We do not know beforehand at what stage (i.e. after how many terms) the terms of the series will become negligibly small, and it would be a waste of time to compute 20 terms if the last 15 or so made an insignificant contribution to the sum.

The programmer defines how the machine is to compute a series term by term, and in addition he lays down the criteria which specify the precision he wants. The machine continues to compute successive terms in the series until the addition of the last term it has calculated makes a difference to the sum which is less than the specified amount. The machine then stops computing the series, assumes that the value which it has obtained is correct and goes on with the next part of the calculation.

If this example seems to be rather academic, let us anticipate for a moment the discussion in Chapter 20 on engineering design. The stresses in each member of a hypothetical girder structure are computed; if at any point they exceed the safe limit the machine will automatically modify the proposed design and strengthen the weak point.

As our last example we show how the *B*-tube can now be used to help the programmer in other ways which Lady Lovelace suggested. Suppose that one wants to raise a number x to the power n, where n is an integer, which may have been computed by the machine, or may alternatively have been specified by the programmer. The number x is put into the mill, and simultaneously n is put into an auxiliary store, which may for convenience be the B-tube. x is multiplied by x and unity is subtracted from n. The process is continued, always multiplying the previous result by x, until the value of n has been reduced to unity. By this time the machine will have worked out x^n, and can proceed with the main computation. This is a very simple example of the way in which the machine can, as it were, "compute" its own programme. Other examples will appear in Chapter 5.

It is important to stress that the criteria which the machine has to observe are specified by the programmer, but the choice is made by the machine, which in this way is able to control its own operations in the light of the results which the computation has yielded.

The process of making a choice is in fact just as automatic as the operation of a tram car which goes to left or right according to the manner in which the driver operates the points. The machine makes the decision in the following way. The number whose sign is to be determined is presented to an *and* gate, to the other terminal of which goes pulse $P20$ (or whichever pulse coincides with the most significant digit of the number). We have already explained that a negative number has 1 as its most significant digit. The *and* gate will produce an output pulse if the number is negative. This output pulse trips a flip-flop which is made to choose the number of the next instruction which the machine obeys. It is usual to arrange that the serial number of the next instruction is either "last number plus one" or "different number specified in the instruction." The flip-flop opens gates which put one or other of these two numbers into the control tube, and the machine carries on.

It is clear that the machine will be capable of performing the same series of arithmetical operations on any set of numbers presented to it. The programme itself is conceived in purely general terms, and can be written throughout in symbols. To quote once more from Lady Lovelace, who in this passage is referring to the cards on which the programme of the machine was to be punched: "The cards are merely a translation of algebraic formulae, or to express it better, another form of analytic notation. . . . The cards only indicate the nature of the operations to be performed, and the [address] of the variables on which they are to be executed, the cards themselves will possess all the generality of analysis, of which, in fact they are merely a translation . . . the engine weaves algebraic patterns just as a Jacquard loom weaves flowers and leaves. The machine is, in fact, the mechanical and material representation of analysis."

This is why Babbage called it an *analytical engine.* His difference engine was capable of only one type of computation, as a special programme had been built into it. The analytical engine, because its programme was to be on cards, would have infinite flexibility.

Babbage's "squares" problem can be used, as he saw, to demonstrate one of the limitations of the machine. Suppose we had asked it to find the smallest integer whose square ended in 269697. Now we know, but the machine does not, that there is no integer whose square ends in 7. If we were to pose this problem to the machine in the way which we have just discussed, the poor thing would go on trying until it broke down or wore out. The same thing would happen if we tried to make it design a bridge to satisfy impossible conditions.

Many years ago we made out of half a dozen transformers a simple and rather inaccurate machine for solving simultaneous equations—the solutions being represented as flux in the cores of the transformers. During the course of our experiments we set the machine to solve the equations—

$$X + Y + Z = 1$$
$$X + Y + Z = 2$$
$$X + Y + Z = 3$$

The machine reacted sharply—it blew the main fuse and put all the lights out.

Unfortunately these digital computers have no similarly emphatic method of registering their opinion of the programmer, but under certain conditions they can be made to blow a hooter if something has gone wrong, and from the character of the note which the hooter produces, a skilled programmer can often tell what ails the machine, much as a mother understands the cries of a very young child.

The hooter gives a single "click" every time the machine encounters the *H*-instruction so that by including *H* in a subroutine* we can tell by the pitch of the note which the clicks produce how frequently the subroutine is followed.

When the Manchester University machine has completed one testing programme which it needs when playing the game of draughts, it prints out "That's all now" on the teleprinter, and plays "God Save the Queen," in the key of B flat.

* See Chapter 5.

Alternatively if we exploit the random number generator we can make the machine produce a series of notes, which may be selected at random from the semitones of the conventional scale, or may alternatively have an entirely random pitch. The "music" which results is unique in the sense that nothing exactly like it has ever been heard before, or will ever be heard again. It is reminiscent of the works of some composers who should, perhaps, not be identified by name.

In December, 1951, the B.B.C. broadcast the machine's performance of "Jingle Bells On a One Horse Open Sleigh," and "Good King Wenceslas." This must be the most expensive and most elaborate method of playing a tune that has ever been devised. The whole computer is used in the operation, and, let us repeat, it does not merely produce successive musical tones, as an ordinary instrument does, but each separate vibration has to be individually programmed. On the whole it seems probable that this technique will have no other applications than the amusement of the programmers and their friends at Christmas time.*

On the other hand Lady Lovelace pointed out that: "If the fundamental relations of sounds in the science of harmony and musical composition were susceptible of adaptation to the notation and mechanism of the Engine, it might compose elaborate and scientific pieces of music of any degree of complexity and extent." Some day we must make our machine orchestrate one of the tunes it has composed.

In spite of the fact that he can exploit the judgment of the machine, the programmer still carries a great responsibility. The machine will slavishly perform all that he asks of it, and it is up to him to make sure that the questions which he poses are sensible. If they are not, he may be very surprised at the answer he gets. This may seem trivially obvious, but, in fact, it is very difficult for any one to take a proper "machine's eye view" of the operating instructions, that is to say to look at them from the point of view of the machine which can only follow them literally, and to make sure that the machine will be provided with instructions to cope with each and every possible eventuality which may arise in the course of a calculation.

A hundred years ago Samuel Butler wrote prophetically in

* The "Whirlwind" computer at M.I.T. (see page 177) is much more highbrow than this one. It plays Bach Fugues at Christmas time, but everyone *recognizes* the tunes played by the Manchester machine.

Erewhon: "Machines have their own tricks and idiosyncrasies; they know their drivers and will play pranks on a stranger." Most 'prentice programmers find that they have to live with their machine for months before they are admitted to the circle of its intimates, and anyone who, presuming on a long acquaintance, attempts to programme without all due care, will find that the machine will take advantage of every slip to trip him up.

The art of programming is rather specialized and an example would, therefore, be out of place in this chapter. On the other hand the story is incomplete without at least one simple example, which is to be found in Chapter 5.

Most people find programming rather difficult at first, and it involves as much effort as learning to play chess, or to speak a foreign language. Beginners are perplexed by rules which seem as arbitrary as taking a pawn *en passant*, or as the conjugation of verbs which are irregular to the point of impropriety in their behaviour; we have found that an ordinary mathematician can learn enough to programme straightforward problems in a month or two, though of course greater experience brings greater skill and fluency.

There are large numbers of standard operations which are likely to be required quite often, for example, the decimal-binary conversion which is needed every time a number is put into the machine. Other important examples are the computations of the sine of an angle, and the logarithm of a number. If routines are made for these operations they may be put into the machine every time they are needed, once a standard tape or deck of punch cards has been prepared; alternatively, such routines can be stored permanently on a magnetic wheel and called down into the high-speed store whenever they are needed. All that the programmer has to do is to punch the address in which the routine is stored into his main programme.

A great deal of work has been done, particularly in Cambridge, to simplify and standardize methods for constructing programmes out of subroutines. The subject is discussed in detail in a book* and in outline in the article on the E.D.S.A.C. (Chapter 6).

* See page 113.

Chapter 4

THE CONSTRUCTION, PERFORMANCE AND MAINTENANCE OF DIGITAL COMPUTERS

His faults were such that one loved him the better for them—

GOLDSMITH

SO FAR WE HAVE NOT ATTEMPTED TO ANSWER what must be the most important of all the problems which confront the potential user of a digital computer, who wants above all to know how many results he may reasonably expect from a given machine in the course of a working day. It is the purpose of this chapter to throw some light on this question, and we shall have to discuss not only the design of the machine, in general terms, but its probable reliability, as well as the type of problem which it may be asked to solve. First of all let us consider the fundamental question of the reliability of one of these complicated machines. Is it possible that a machine containing three or four thousand valves, ten thousand resistors, and a hundred thousand soldered joints, can ever be made to work, or that it will go on working reliably once it has been finished?

The life of an ordinary radio valve is usually assumed to be about five thousand hours. Experience has shown that ordinary electronic equipment as used, for example, in nuclear research suffers about twice as many faults from components such as resistors, condensers, transformers, etc., as from valves. This equipment is of the portable bench-operated type and it is exposed to the ordinary hazards of laboratory use, i.e. it is switched on and off at irregular intervals; the mains voltages may fluctuate; it may be shaken and perhaps dropped, and people may occasionally spill cigarette ash or cups of tea into it. No particular precautions are taken to keep it cool.

If one assumes that similar results are to be expected in a digital computer containing 3,000 valves, one would expect a fault of some kind every half-hour. It is not easy to find and clear a fault in less than half an hour, so that one might expect that the utmost efforts of the maintenance crew would just fail to make the machine work at all. Fortunately the pioneers were not worried by figures like these; experience has shown how right they were to be undeterred by statistical evidence based on a few selected truths. Modern computers do in fact work very well, but it was by no means obvious that they would.

We have explained that pulses circulate throughout the machine at the rate of several million per second, and that, if any single one of these pulses is misinterpreted, the machine will make a mistake. No other type of electronic equipment has to satisfy such drastic conditions. It is quite normal, for example, for a radar receiver or a television set to pick up a few hundred spurious pulses a second, though the operator may not observe them. A few clicks pass unnoticed in a broadcast receiver. An automatic telephone exchange is made out of relays and switches which are simple in construction and which have been under development for half a century; nevertheless no one is surprised, and few complain, if they get one wrong number out of a hundred. This kind of thing simply cannot be tolerated in a computer.

Elaborate precautions are always taken to find and eliminate mistakes made by human computers. The speed of the machine is so great that it can do a thousand times as much work as a man. Is it reasonable to expect that it should be a thousand times less prone to error? It is only undetected mistakes which really matter: if the machine were to go wrong occasionally, one could use the normal checking procedures which are familiar in accountancy and scientific work. Few men make less than one or two mistakes in a day's work, but they find and eliminate them as they go. Let us remind ourselves that if the machine made a mistake once a second, it would be doing as well, relatively speaking, as a man making an arithmetic mistake every few hours. No engineer would be content with such a standard of performance in his machine, so we must analyse the reasons for error in a computer and see how many of them can be eliminated. The machine will fail to give the right answer whenever a component fails, when a valve fails, when a relay sticks, or when, for any reason, the circuits behave in an anomalous way. We shall take these points in turn.

First of all, let us consider the possibility of stray pulses getting into the machine from outside. They can be carried in from the mains, or by radiation from other electrical machinery in the neighbourhood, and they can be more troublesome and insidious than one would expect. We have observed transients on the 50 cycle mains which range from the total suppression of a couple of cycles to stray 10 μsec pulses, 100 V in amplitude. Superimposed on these are the voltage changes due to "load shedding." Many of these transients come roaring through the smoothing circuits of an ordinary power supply and cause trouble in the computer. It is probably best, therefore, to isolate the computer from the mains completely and to feed it with its own local supply from a motor generator set, which can in practice be driven from the mains. Radiated interference is more easily eliminated. The sensitive parts of the computer must, of course, be carefully screened, and it may be necessary to put the whole computer in a screened room if, for example, it is near a radar set or a high-power X-ray machine. A standard Hollerith card-sorting machine is one of the most vicious generators of interference that we know and should never be in the same room as a computer. An American Army headquarters, equipped with sensitive short-wave receivers and card-sorting machines, moved several times in search of electrical quiet before it discovered that it was taking its own "noise" around with it.

The pulses which convey the information have to be greater than a certain minimum size before they are counted. Now it is well known that the random motions of electrons produce a fluctuating voltage across any resistance, and theory shows that, if we wait long enough, random fluctuations will get past any gate, however high it may be; in other words, random noise fluctuations of any size are possible. The equations which govern the production of random voltages are similar to those controlling the motion of the molecules of a gas. In theory, any molecule in the atmosphere might acquire a velocity sufficient to carry it outside the control of the earth's gravitational field. The theory shows (for nothing is more practical than a good theory) that the time which one may have to wait for a large voltage pulse to occur is very long, so that, if the gate is set high enough, the chance of a random pulse being counted is very small. The earth's atmosphere is stable for the same reason: each molecule has to wait for a very long time for the chance to escape from the earth, though it would escape from the moon in a few years.

It is clear that the chance of a noise pulse being important is greatest where the signal level is smallest, that is to say at the output of one of the "memories." Exact calculation is difficult; the cathode-ray-tube store is probably the most likely to give trouble from this cause. The random thermal noise at the grid of the first valve of the amplifier is a few microvolts; the output of the cathode-ray-tube "memory" is about a millivolt, and it can be shown that spurious random noise pulses should not cause a mistake to occur in the Manchester University machine more often than once in several thousand years.

If one makes a simple amplifier for a Williams memory, however, one may find that spurious pulses occur several times a second. Electrolytic smoothing condensers often produce large numbers of pulses; many types of coupling condenser seem to give a pulse a second; composition resistors and dry-soldered joints which are carrying current may produce occasional pulses of more than a tenth of a volt. The use of high-quality condensers and wire-wound resistors reduces the number of spurious pulses to about one an hour; these are due to the first valve. Most high-slope pentodes seem to cause trouble of this type for reasons no one understands. Research to eliminate spurious pulses from amplifiers used in nuclear physics at Harwell has shown that one particular valve (the EF37A), which has a medium slope and is specially free from microphony, is quite free from spurious pulses. A good amplifier should produce not more than one pulse a week, of size comparable to the amplified millivolt pulses which it receives from the store.

The resistors and condensers out of which the machine is made always differ to some extent from their nominal values, and change with the passage of time. It is therefore necessary in designing all circuits to make due allowance for these changes.

It is reasonable, for example, to use components which are within plus or minus ten per cent of their nominal value, and to assume that they will vary by not more than 10 per cent after installation. Most circuits can be made to work if the resistors are within 20 per cent of the specification. High-stability components can be used where necessary, but a binary machine is very tolerant of component variations, and the number of preset variables should be reduced to a minimum. It is nevertheless desirable to under-run everything and keep all components cool by blowing them with dust-free air. If this is done, it is found in practice that components last for years without giving trouble. For example, it has been necessary to change only two resistors in the Manchester machine in a year. It is almost unnecessary to add that the construction of the whole machine should follow the best known engineering practice, but it is worth pointing out that, if ordinary television sets were made in small quantities with the same care and with all the precautions which are conventionally used in building a computer, the chassis would cost more than £500 each to make.

Some of the precautions which are necessary are not by any means obvious, because the requirements which a computer imposes on some components reveal faults which ordinary experience would never show up at all. For example, the coating of the cathode-ray-tube screen which is used in a Williams "memory" must be completely homogeneous. The store depends for its success on the constancy of secondary electron emission all over the surface of the screen. Ordinary screens are made out of what appears to a layman to be a mixture of ground rock and extract of seaweed—all very carefully purified of course. The coefficient of secondary electron emission of such surfaces is quite unpredictable although they glow quite nicely when electrons hit them. The tubes which are now used in Williams memories have been specially developed by the research laboratories of the General Electric Company, Wembley, which devoted some years to the production of completely uniform screens. It is because the blemishes which are still to be found in the screens are very small that the use of a relatively large defocused spot for storage is such a help. Only a few per cent of the G.E.C. tubes have to be rejected for blemishes these days, but less than half of them are good enough to be operated using the double-dot system originally suggested by Williams and Kilburn (see page 61). Similar precautions are necessary with the magnetic drum; its nickel surface must be completely free from pin-holes and have a uniform microcrystalline structure. The best ordinary commercial electroplating is done for the preparation of matrices for gramophone records because a single pin-hole would produce a "click." All possible precautions have had to be taken to improve on available technique in order to produce a coating which will store digits on all parts of its surface. When the time comes to develop stores on ordinary magnetic tape of the type used in sound recording it will probably be impossible to rely on complete uniformity of coating, and it will be necessary therefore to test the tape as it is used, probably by reading back what has just been written, and re-recording a few digits if there is any discrepancy between the record and the information in the machine. This process is quite easy if one records at a speed significantly slower than the normal digit speed of the machine, as one can do by making use of a cathode-ray-tube store (which is an aperiodic device) as intermediary between the machine and the tape.

The chief source of trouble in most machines is valve failure of one sort or another. A machine may contain several thousand valves, all of which must be working properly if the machine is to function, and if we are to understand the way in which it will behave, we must recapitulate very briefly some of the work which has been done during the past few years on the lives of valves.

A valve may fail catastrophically, e.g. its heater may go or it may develop an internal "short." Most valves, however, seem to fail gradually. It follows therefore that the useful life of a valve depends upon the deterioration in its performance which can be tolerated in a particular circuit before it fails to operate. One circuit may work when the emission of the valve in it has dropped to one-tenth normal; another may fail when the emission has fallen by a half. A valve in the first circuit may have twice the average effective life of one in the second. In general it is hard to design a very fast machine which will work if the valves deteriorate far; a certain minimum current is necessary to charge up stray capacities in a short time so, for this reason, speed has to be bought at the expense of reliability. A slow machine can be made very reliable. For example, Nimrod (see page 286), which had 500 valves (type 12AT7 made by S.T.C.), ran for five months (= 2,000 hours) with only two valve failures. After it had worked throughout the Festival of Britain it was dismantled and taken by air to Berlin. It was working twenty minutes after it was first switched on, and ran for the three weeks of the Berlin show with no failures at all.

In spite of the fact that it is impossible to speak of a unique "life" for a valve, a great deal of statistical work has been done, particularly in America, to study the failure of valves when they are used in large numbers. It is usually found that if the number of survivors is plotted against time, the curve is roughly exponential, very much like the curve which one gets if one plots a similar graph for surviving radioactive atoms. This result is rather surprising, and suggests that the expectation of life of valves which are still working is almost independent of the life which they have already lived.

This is only a first approximation, and typical results vary greatly. A certain repeater valve has been found to have an almost constant expectation of life of about 100,000 hours for a total period of 140,000 hours, or nearly 16 years. A typical American-made receiver valve (the 6SN7) had a probable expectation of life which varied from 6,000 hours to 4,000 hours during 20,000 hours' use in one large American computer during the years 1948–50. Other valve types such as the 6AC7 and the 6L6 had lives of about the same duration. These figures, which were announced at a convention in Atlantic City in December, 1950, are representative of the experiences of most American computer designers. They suggest that no useful purpose would be served by changing all the valves in a computer after a fixed length of time. It is better to replace valves as they fail or show signs of imminent failure. A batch of new valves will, on an average, last no longer than a similar batch which may have already done several thousand hours' service. We have been told that four out of a team of six engineers who are building a computer in one midwestern city in America were hit by spent bullets during their first year's work. Such a process is apparently random, and one might reasonably expect that the number of surviving computer engineers in this particular group would decline exponentially—but why should valves behave in the same way?

Several causes of valve failure have been established; bad workmanship may cause a batch to fail quickly; small pieces of lint may carbonize and, at some stage, fall across between electrodes, particularly between grid and cathode. Care in production will minimize such troubles. Valves can be selected by shaking them while testing for high inter-electrode resistance. Some valves go "soft," some lose their emission, in others a high resistance forms between the nickel sleeve and the oxide coating of the cathode. To help to activate oxide cathodes, it is customary to add a small amount of silicon to the nickel of which the cathode sleeves are made. After a time this silicon forms a high-resistance compound between the nickel and the oxide. This layer is destroyed by electrolysis so that an interface layer of large resistance forms most quickly in valves which are switched on but pass no plate current. A valve with an interface behaves as if it had an impedance in its cathode lead, like that shown in Fig. 4/1. The rate at which R increases is shown in Fig. 4/2. It has been found possible to activate cathodes successfully with less silicon than is normally used, and the layer does not seem to form at all if the nickel contains less than 0·01 per cent silicon.

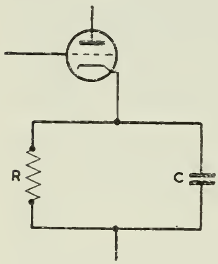


FIG. 4/1.
The virtual impedance in the cathode of a valve due to the layer which forms between the nickel and the oxide coating
The time constant RC is about one microsecond

Most British-made valves use magnesium instead of silicon as an activator, so that this form of failure, often called "sleepy sickness," is almost unknown on this side of the Atlantic.

It may surprise most people to be told that no really satisfactory valve holders have ever been made. After some years a resisting film seems to be built up between the valve pins and the holder, so that Post-Office engineers occasionally go round repeater stations and move the valves in and out of their sockets to break up the film and re-establish electrical contact. This source of trouble is minimized by using silver-plated valve pins. Moreover, the process of pushing the valve into its socket often strains the glass and produces minute cracks which make the valve go soft after a time.

Long-life valves are now being designed, both in England and in America, which will be most carefully built, will use silicon-free

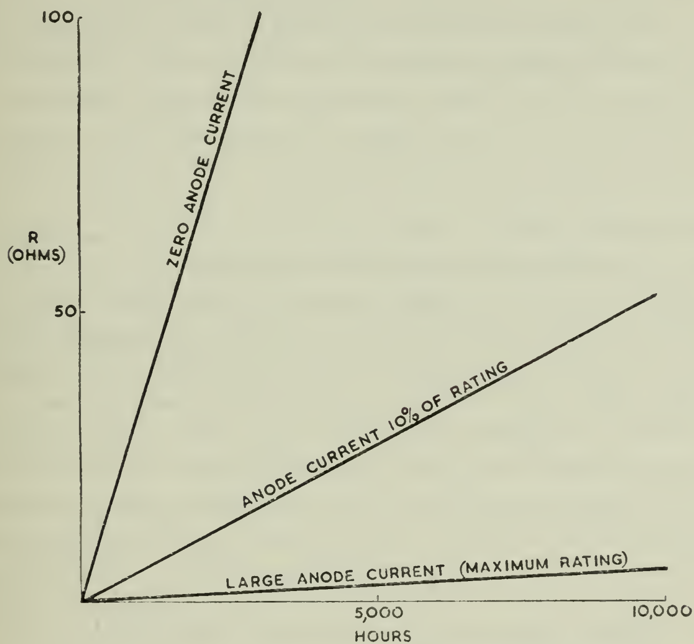


Fig. 4/2. Graph showing growth of the interface resistance R in a typical valve for maximum rated anode current, ten per cent of maximum current, and zero current

cathodes, and will be fitted with flying leads which can be soldered directly into their circuits. No figures are yet available for these valves, but it is hoped that no more than 1 per cent will fail in the first thousand hours' use; this would amount to one tube every thirty hours in a typical computer.

Meanwhile, the rather old-fashioned English-made valves which are in use in modern computers over here are giving very good service—far better in fact than one would have expected; the latest figures for the E.D.S.A.C. and M.A.D.M. are of the order of one or two tube failures per week and both machines have run for several weeks with no valve failures at all. When first these computers were put into service, the failure rate was far higher, but once the defective valves had been removed the figure settled down quite quickly to a rate of about 1 per cent per thousand hours or better. This result is about four times as good as the valve manufacturers expected, and suggests that the average valve life may be 100,000 hours or more. It is too early to be sure if our valves do in fact "decay" exponentially; such results as we have seem to indicate that, if anything, they become more reliable as they get older, and that the best guarantee that a valve will run for 10,000 hours is that it is still working after 10,000 hours' continuous use. The E.D.S.A.C. in Cambridge has now been running for more than three years, and the valve failure rate shows signs of dropping below 0·5 per cent per thousand hours.* Our experience does at least show why relatively few reliable statistics on valve life are available in this country. It takes much longer to collect them than it has done hitherto in America. It remains to be seen if the new valves are any better than those which we are now using—most of them EF50, EF55, EA50 and VR116—which were designed about fifteen years ago, were used in millions during the war, and are now considered to be obsolescent.

A great deal can be done to minimize trouble due to valve failure. Most computers are given a daily check of all fundamental waveforms, so that a valve which is nearing the end of its useful life can be found and removed before it has declined so far as to endanger the working of the machine. Furthermore, suitable test programmes can be devised which will check all the operations of the machine in turn and may, under certain circumstances, indicate in code the particular chassis, and even the particular valve, which has failed. The most difficult faults of all to trace are those which are due to a valve only just on the verge of working; such faults are often intermittent, and they can sometimes be made permanent and easy to find by putting attenuators in some of the leads, or by a systematic routine of varying all the filament and plate voltages up and down while the machine runs a test programme.

The technique of *preventive maintenance*, as it has been called, has been studied intensively in America, and its use has considerably reduced the number of valve failures which occur while the machines are running. We hear that the latest figures for big machines using modern tube types are of the order of 1 or 2 per cent per thousand hours, though the crude failure rate is still considerably higher. Preventive maintenance has not been so elaborately organized in

* During the latter part of 1952, the replacement rate for valves in the Manchester machine dropped to a little more than one-half of one per cent per thousand hours.

this country, and so perhaps half of all valve failures occur during computing time, particularly during the first hour or two on Monday morning after the machine has been switched off for the week-end.

Under good conditions a machine which is properly looked after will run for more than 95 per cent of the time. Figures of 100 hours in a week are commonplace, and several machines have run for 160 hours per week for several months.

Most people would imagine that relays and switches and similar electromechanical devices, which are easy to understand and have been undergoing intensive development and study for a hundred years or more, would be the most reliable parts of a computer. On the contrary, they cause proportionally more trouble and are probably less reliable than any other part of the machine. It is therefore essential to check their operation in some way and this is usually done by reading the position of the device electronically and checking back to see if it is as called for by the electronic instruction.

Electromechanical devices such as relays tend to suffer much more than valves do from intermittent faults of one kind or another. For example, the contacts of a relay may pick up a little dust which causes them to fail once or twice or even half a dozen times in succession before the dust falls out. This type of fault is particularly exasperating because it may make a computer produce the same incorrect answer two or three times in succession and thereby defeat some of the more normal checking procedures, but there is nothing for the maintenance crew to look for and cure. The commutator of the rotating machine which provides the main power supply for the computer may wear and arc intermittently, thereby generating transients. The commutator "flash" may even be caused by a speck of dirt; it may last for a very small fraction of a second, but it may generate a hundred watts of radio frequency energy, some of which gets into the computer and causes a mistake. Phenomena of this type may be so infrequent that they would be almost undetectable in any other kind of electrical equipment, they would do no harm at all, for example in a radar transmitter or a television set, but they can be responsible for most annoying errors in a long computation. Every part of a digital computer, including all auxiliary equipment such as the input-output mechanism and the power supply, must be very good indeed if the machine as a whole is to be of any use at all.

Some form of check is very useful in both input and output mechanisms, which involve mechanical moving parts and are more likely to fail than the computer itself. For example, if numbers are put into a machine from standard five-hole teleprinter tape, one should arrange that all the symbols which represent the digits from 0–9 contain two holes, so that a single error in the punch cannot convert one number into another. A mistake in a programme is much less serious than a mistake in a number as it would almost certainly be found and eliminated by the programmer. If the machine uses seven-hole tape and all meaningful characters in the code use three and only three holes, 35 combinations are available, and very few mistakes will arise from the coincidence of two errors in the punch. This scheme is used in certain long-distance radio-operated teleprinter links which convey information by combining three out of seven available radio frequencies. It is in any event desirable to check that the correct typewriter key is depressed by each character on the tape which operates it, and the whole output system can be "verified" by punching two tapes, using one to operate the typewriter, and the second to verify the signals which are transmitted back from the typewriter when its keys are depressed.

When all precautions have been taken, a computer should be capable of many hours of operation without error, but nevertheless it is worth while exploiting the speed of the machine and including in the programme all the checks which a man would apply if he were doing a similar computation by hand. Certain calculations can be repeated in a different way; for example, multiplier and multiplicand can be interchanged and numbers added in a different order. Complete accuracy can be assured at a slight sacrifice in overall speed, a sacrifice which is usually well worth while.

If, for example, the machine makes a mistake on an average once an hour, one should, ideally, make it check itself in some way or another perhaps once a minute, so that if it does find something wrong and has to repeat part of a calculation, it will, on an average, lose less than one per cent of its time in this way. It is probable that the skill of a programmer is shown more clearly by the way in which he coaxes reliable results from a fallible machine than in anything else that he does. One must not lose sight of the fact that if he makes it check its performance once every ten minutes, and if the machine makes a mistake every five minutes for some reason or other, it may misinterpret only one pulse out of several hundred million—and produce no answers at all. This checking procedure can be handled effectively only if the machine has a large memory, for example a magnetic drum, on which answers can be kept with complete confidence for an indefinite period. The machine can be made to keep a record of the time which is spent in correcting its own mistakes; it is not unreasonable to expect that the total time spent in this way should not exceed a few per cent at most of the total time of a computation.

The precautions which should be taken depend to a great extent on the nature of the problem which is being solved, and on the type of mistake which the machine is likely to make. For example, it is known from experience that certain machines run correctly for hours at a time, *or* they make an occasional mistake of a single digit, *or* they make catastrophic mistakes and perhaps stop working altogether. Other types of error are almost unknown. A certain amount of care on the part of the operator will allow him to be certain of his results. If for any reason there is an intermittent fault which causes an occasional digit to creep into the stores, it is as likely to cause errors in the instructions as in the arithmetic. The results of a mistake in the programme are usually obvious. Mistakes in the most significant part of a calculation are usually obvious on inspection; and mistakes in the least significant digits can usually be ignored, so that if all the results look reasonable it is very probable that they are all correct. Errors in the arithmetic unit can usually be detected by test programmes, and so it follows that in many calculations in which a very few undetected errors are unimportant, it may pay to exploit the speed of the machine for all it is worth and check the calculations only by repeating every tenth computation as a test for consistency of operation of the machine. In other computations which would be ruined by a single error it will be necessary always to repeat every calculation as well as using all the conventional mathematical checks such as "differencing" the results.

Machines have been designed with built-in checking facilities. The U.N.I.V.A.C. which was built by Eckert and Mauchly in America, consists of two complete machines which simultaneously perform the same calculation and check each other's results at each step. Other machines have been built which perform the equivalent of "casting out nines" on each instruction or each number. For convenience, since they work in binaries, they cast out threes, sevens or fifteens. The residue after casting out is stored and checked against the value computed every time the instruction or the number is used, and arithmetic operations are checked in the normal manner familiar in casting out nines in conventional arithmetic. The accuracy of the machine can in this way be verified at the price of additional complexity in its construction. The system assumes that two mistakes which compensate for each other are unlikely to occur simultaneously in the same number, and it remains to be seen if the extra components really justify themselves in use.

Another type of checking system is described in some detail in Chapter 10.

There is one type of error which is sometimes rather insidious, but it can be dismissed after it has once been explained. This is the error due to "round off." It occurs in all ordinary arithmetic, but it assumes an unusual importance in computing machines because of the length of the computations which they perform. If two ten-figure numbers are multiplied together, either on a desk machine or by logarithms, they may produce a product which has only ten figures, whereas if the work were done by hand the answer would contain twenty figures. The error which is introduced by neglecting the last ten places is of the order of "half" one of the least significant digits which is retained. If the computation involves a million such multiplications the errors accumulate in random fashion and amount to 500 times one of the least significant digits. In ordinary work, errors of this type can be ignored if one computes to one or two more significant figures than one needs in the final answer, but the calculations done on a digital computer may be in error unless they are done to ten or twenty more binary digits than the final answer. This is of course perfectly straightforward if only one remembers to do it.

The machines have already reached the stage when the most common source of error is the human beings who programme them. Certain obvious precautions can be taken. The utmost use should be made of subroutines which have been previously checked and are known to be correct. Tapes should be punched twice—perhaps by different people—and a master tape made automatically by comparing the two tapes on a machine which stops punching as soon as it detects a discrepancy. Few people realize the care which has to be taken to proof-read tables of figures. It is quite usual to insist that a table should be compared against the master copy by five people independently in succession before it is accepted as error-free. The problem of checking tapes is quite as difficult. An elaborate editing system should be used to find and eliminate the majority of errors on input tapes or cards before they are put into the machine. Babbage was well aware of the risk of error in transcribing figures and he designed a system by which his Engine was to set up its own results in type; the type itself was to be checked automatically to ensure that each container in the type-setting machine held only the correct figures (e.g. all sevens) so that when a 7 was demanded by the machine a 7 would appear and the machine itself would have verified that it was indeed a 7. Most modern machines print out their results automatically on a typewriter of some kind in a form which can be reproduced photographically by a lithographic process. If the machine punches out its results on tape, this can then be made to operate an automatic typewriter or a monotype machine.

Some human errors in input are inevitable but there should be none in the final printed output of the machine excepting those directly attributable to mistakes in input. As far as possible therefore, input data should be encoded automatically. For example, if the machine is to analyse experimental results derived from indicating instruments, the meters themselves should be made to present their readings directly to the machine, thereby avoiding all human errors in transcription.

We can derive a certain amount of information about the difficulty which human beings have in manipulating numbers from the history of automatic telephone exchanges. When first the Bell System proposed to change from manually operated switchboards to automatics they did some experiments to estimate the probable efficiency of the system when it was used by the public. They were proposing to use seven-figure numbers such as 3242271, and they found in practice that the number of mistakes in dialling was so big that they were on the point of abandoning the idea. At this point someone suggested that a few of the numbers should be replaced by letters, associated if possible with the name of the exchange, so that the same series of impulses was produced by dialling FAI2271—the public could do this, so the automatic telephone became possible.

In one elaborate computation which was recently performed in America all the input data was prepared in the following way. It was punched on Hollerith cards. Each card was verified by being repunched by another operator on a special punch which detects discrepancies. A second set of cards was punched and verified. The two sets were compared (automatically by one special machine which rejected faulty cards). The approved set of cards was then printed out, and the results compared with the original documents by five independent proof readers. In this particular calculation a single undetected error would have been a catastrophe—and it may serve as an awful warning of the lengths to which people may be driven in order to be quite certain that the machine is supplied with the data it really wants.

It may help the reader to appreciate the difficulty of punching a tape which is free from error if he realizes that it is comparable to typing out a long script in a foreign language, in which a single spelling mistake turns the whole thing into gibberish.

Finally, when all these precautions have been taken, it must still be assumed that most programmes will be wrong when they are first prepared. A systematic procedure must be followed to find the errors. Two or three methods are worth mentioning. The first is to make the machine print out the contents of the accumulator at certain points in the calculation so that the results can be compared with those obtained from hand computations. It is important to do this over a representative range of parameters: we recall, for example, a routine which worked out the logarithms of the first 45 integers correctly, but was wrong thereafter—the mistake in the routine took hours to find. The second method is to make the computer print out the successive operations which it performs (add, multiply, etc.), in order. A third is to make it stop at certain points so that the operator can visually check the contents of each store. This is to be deprecated because it takes up machine time uneconomically. An even more wasteful procedure, which is sometimes inevitable, is to drive the machine by *manual prepulses* so that it goes through the successive stages of a routine slowly enough for the operator to watch the process in detail. When the programme is known to be correct, it is run through at full speed, and these checking procedures are facilitated by the use of a *dummy-stop* instruction which stops the machine if a switch on the console is *on*, but otherwise has no effect.

When all this has been done, two questions remain: "Are the original equations right and do they fairly represent the problem?" and "Was the problem worth solving anyway?" We are seldom as certain as we should like to be about the answer to either, and it is embarrassing for an engineer to discover after all his efforts that he has produced what Pirandello might have described as *Six Solutions in Search of a Problem.*

The rate at which a machine handles a problem depends on many factors, some of which are matters of logic and design, and others of routine engineering. In the first place, the overall reliability of the machine which has already been discussed has an enormous effect on the rate at which it will handle a problem of reasonable length. It is, of course, necessary to break down the computation into small steps, each of which can be independently checked. The less reliable the machine, the shorter the steps must be, and in general, the more complicated the checking procedure. This means that even when it is working well, a machine which is thought to be unreliable will, other things being equal, do less work than a similar machine which is known to be reliable.

Let us consider what determines the speed of a machine which is known to be in good adjustment, and which makes a number of mistakes which is too small to worry the programmer. In the first place one must remember that a machine is capable of performing only a very limited number of types of operation, and that all problems have to be broken down into exceedingly simple steps before the machine can handle them at all. This means that a machine has to perform many more operations than a human being would go through, if he were doing the same computation. Its relative incompetence is due primarily to the fact that it cannot take an overall view of a problem and select the significant item on sight. This is particularly well shown, for instance, if the machine is made to play chess. Chapter 25 explains how badly the Manchester machine does this compared with a human opponent. It plays a reasonably good game of draughts, but its chess is very poor indeed, and this in spite of the fact that it can add up a thousand numbers in a second. The question which we have to answer is: "What sort of a performance can we expect from such a machine if we ask it to undertake any ordinary arithmetic calculation?" It must be clear to the reader that there is no unique answer to this question, but it is perhaps worth while anticipating our later discussion and saying that we usually find that the Manchester machine handles representative problems at a speed between 200 and 1,000 times faster than an ordinary fairly skilled computer who is using a standard desk machine. We use the factor "500" in preliminary estimates.

It is clear then that the nature of the problem and the skill of the programmer make a great difference to the output of any machine. The designer can do much to lighten the burden of the programmer, for it is he who determines the total storage capacity of the machine and the number and variety of the operations which it performs per second. A machine which has a small memory, every part of which is available in a few microseconds, may be able to work exceedingly fast until it has done everything it can think of for the moment, but unless it restricts its operations to very simple problems, it may then have to wait so long for fresh information that its overall speed may be quite low. A large memory is essential if a machine is to handle complicated problems; many scientific problems may require the use of a hundred thousand digits and some commercial work involves millions, and unless all this information is available almost instantaneously on demand, the machine may be quite slow, however quickly it can add two numbers together. It is for this reason that large memories on magnetic drums and magnetic tape form an essential part of all big machines. The skill of the programmer is shown by the way in which, for instance, he can arrange that information which he will need is made available on a part of his tape next to the bit he is already using. This problem does not arise if the information is written on a drum, but the capacity of a drum is smaller than that of a tape, and it may well be necessary to make some future machines with both tape and drum stores.

A rather important disadvantage of a machine which has a small high-speed memory may not at first sight be as obvious as the others. It is one of the operational weaknesses both of the A.C.E. (see Chapter 8) and the C.P.C. (see page 177). Both of these machines take in their programme by using punch cards, and it frequently happens that the whole of a complicated routine is too big for the fast memory. A great deal of the time and ingenuity of the programmers is sometimes involved in tailoring their programmes to fit the machine, but when this proves to be impossible they may have to reproduce some of the subroutines several times on punch cards, as neither machine can make the cards run backwards and use them again once they have passed through. Since it often so happens that the machine makes use of its power of choice to decide whether or not it *wants* to use a routine again, it may be necessary to provide for all eventualities, and to supply the machine with a very large pack of cards indeed, containing a great many duplicates. Of course, it is quite easy by using modern punch-card techniques to prepare duplicate cards, but this is not the point—this system slows down the machine and is obviously a nuisance. Perhaps the most remarkable evidence of Babbage's foresight is to be found in his discussion of this particular problem. The reader will find Lady Lovelace's account on pages 379–95. She seems in a way to be foreshadowing the use of the drum, and this more than a hundred years before either the A.C.E. or the C.P.C. was thought of.*

It is quite impracticable to discuss in detail the effect on the speed of operation of the machine of subtleties in its logical design.

* The P.C.E. machine, made by Samas of Paris, recirculates the cards much as Babbage proposed to do.

There are, however, one or two points of general interest which should be mentioned. In the first place, it is possible to use several different forms of codes. A single instruction, in the Manchester machine or in the E.D.S.A.C., is of the type: "Perform a certain operation on a single number in *one* specified place." The operation may, for example, be: "Add this number to the contents of the accumulator (and put the sum into the accumulator)." A single instruction in another machine might be: "Add the contents of the number in location X to the contents of the number in Y and put the result in Z." These two types of code are called *single address* and *three address* respectively, and it will be clear that a single-address machine does less in one operation. On the other hand its instructions are shorter, so that there is room for argument as to the merit of gaining greater speed at the expense of the greater storage capacity required for the instructions. Most modern machines follow Babbage, and use a single-address code, though as far we know only one (which was made by Bell Telephone Laboratories) incorporates the precaution which he included to reduce the chance of error. He proposed to make it impossible to write into a store unless it contained zero. It is unfortunately easy to write something inadvertently over a number which one wants to keep, but this is just one of many refinements which may or may not be worth while. Another special instruction which he had and which is seldom used nowadays was to write the contents of a nominated store into the accumulator and simultaneously clear the store. It is usual to synthesize this instruction by writing zero into the store in a separate operation, and it is doubtful if the loss of speed really matters.

It is clear, however, that the larger, within limits, the number of special instructions there are in a machine, the faster it will be; there have been never-ending arguments as to the relative merits of different codes, and no two designers have ever agreed on the use of the same set. It has been found experimentally that the use of the *B*-tube speeds up the Manchester machine in the ratio of 3 : 2 in representative calculations. Similarly it has been found that the use of a special multiplier, which multiplies two numbers together in one operation and thereby increases the speed of this process twentyfold, has effectively doubled the overall speed of the machine. It is much more doubtful if it would be worth while to include a special divider,* for it is found that division is needed much less frequently than multiplication. This fact must have been known for

* See Glossary.

many years; as long ago as December, 1663, that experienced computer Pepys wrote in his *Diary*: "My wife rose anon, and she and I all the afternoon at arithmetique, and she able to do additions, subtractions and multiplications very well, and so I purpose not to trouble her yet with divisions, but to show her the use of Globes."

There is no reason to believe that existing machines have achieved anything like the maximum possible speed of computation, even if we restrict ourselves to the components which are now on the market. Speed can be bought at the price of complexity of design, as well as by speeding up the operations of the machine itself. It is probable that a machine which used a parallel store and an appropriate arithmetic unit could do as many as 20,000 or even 50,000 single operations per second. Such machines are already under construction both in England and in America (see pages 144 and 176). A suitable backing store in the form of a very high-speed drum could be built, and such a machine might well have an overall speed more than ten times greater than that of any machine which is now working in England. There is no doubt that even faster machines will be built in the next few years.

There is a real range of problems for the solution of which a tenfold increase in speed would be very useful. For example the integration of one set of differential equations which define the motion of an aircraft took several hundred hours of machine time, and one or two statistical problems have been just as tedious. It is, of course, quite easy to devise problems which are too long and too complicated for any machine to solve. Few people are likely to complain if certain problems (in number theory for example) are as insoluble now as they were a hundred years ago, but one must admit that it is rather humiliating to find that it would take all the machines in the world several years to work out, in detail, the wave pattern produced by a ship at sea.

The vast majority of commercially important problems can be solved perfectly well by a machine which is no faster than many which now exist, and the same is true of many important problems in science and engineering. It is probable therefore that many future machines will be little faster than some which have now been built—they may, however, be more flexible and more versatile. Some of the extra components which will be needed and the way in which they may be exploited are discussed in other chapters of this book.

We have yet to analyse, for we have almost ignored them, the restrictions upon machine performance which are due to the difficulties experienced by the operators who have to prepare programmes for them. It is significant that many machines have spent half their working lives in checking programmes and finding mistakes in them and only perhaps a third of the time in straightforward computation. One can deduce from this the startling conclusion that had the machines been a thousand times as fast as they are, their total output would not have been increased by more than about fifty per cent; in the last analysis the correction of programming errors depends almost entirely on the skill and speed of a mathematician, and there is no doubt that it is a very difficult and laborious operation to get a long programme right.

Furthermore, our experience suggests that a majority of the problems which in practice have to be solved are fairly short, and do not involve long "production" runs; the average programme takes so long to prepare before it is put on to the machine at all, that a staff of between thirty and fifty may be required fully to load a big machine which runs for twenty-four hours a day, on seven days a week. A rough count showed that about 150 machines are under construction in America and England. One sometimes wonders where the programmers will come from.

A machine may do as much work as 500 men, while it is actually computing, but few mathematicians have in fact increased their output more than twenty-fold by using a digital computer instead of desk machines. This is quite good, but it is a much less impressive figure than one might have expected. This relatively small product is easily exceeded, of course, if the machine has a long straightforward run, as it may, for example, if it computes tables, or does commercial work, or if it undertakes a long repetitive computation such as that of the water flow in the St. Lawrence seaway. At one time this seemed likely to occupy all the available computing facilities in Canada for twenty years, but it is now hoped to do it in 500 hours on the new Toronto University machine.

In the course of its lifetime, at least a dozen times as many man-hours will be devoted to programming a machine as were necessary to build it, so that everything points to the importance of designing each computer primarily for the convenience of those who are to use it, rather than regarding it as an exercise for the versatility of the engineers who design it. It is in fact more realistic to ascribe the difficulties of the programmers to weaknesses in machine design and in the technique of programming rather than to human fallibility. In the end almost all machines will have to be handled by relatively untrained staff, and until this can be done as a matter of routine both the machines and the programming technique can be legitimately criticized.

Time has not yet permitted a sufficiently comprehensive study of the relation between the logical design of a machine and the difficulties of an inexpert programmer. Some of the steps which are being taken at present have already been explained. A few others may be of interest. It is probable that the Manchester *B*-tube is important more because it simplifies programming than because it speeds up computation. So far the big memory in the machine has justified itself primarily because no programmer has had to be clever or sophisticated in order to compress a complicated routine and get it all into a small high-speed store. The programmes have therefore been much more easy to produce and to verify than they might have been. A floating point machine, which stores any number as $X \times 2^n$ (where X is between $+ 1$ and $- 1$, and n is an integer), would probably justify itself, not only because it would handle very large numbers, but because the programmer would never have to bother about the possibility of his calculations outrunning capacity and overflowing the stores. In a typical mathematical programme at least two-thirds of the instructions are concerned with non-arithmetic operations—in commercial work the percentage may be even higher—with explaining to the machine, as it were, "which page to read next," "how to organize its data," "where to file things," "how to steer its way through the problem" and "what to do if its pencil needs sharpening." Much yet remains to be done to simplify such operations.

In the next few years something will have to be done to improve the reliability of machines still further, and in particular to make it less necessary than it now is to employ highly skilled men as maintenance engineers. The new "rugged-reliable" valves will probably be much better than any we now have; relay circuits may be eliminated, and completely reliable input-output devices may be marketed. But Ridenour* put the whole problem in a nutshell when he said: "There is nothing wrong with electronics that the elimination of the electronic tube won't fix." A valve is a most inefficient device; it is always necessary to heat its cathode even if it is only called upon to pass current very infrequently; "to use a valve to transmit an occasional pulse is as uneconomic as to send an entire freight train to fetch a single parcel." It will probably always be necessary to work at

* Ridenour, *The Scientific American*, August 1951.

power levels which are much greater than those which would be required merely to ensure that the signal overrides random noise. The characteristics of all available diodes are curved so much that the gates which are made from them will respond satisfactorily only to signals of the order of a volt or more. This means that as long as we continue to use conventional computing circuits, our machines will have to use something like a million times as much power as is theoretically necessary.

The recent development of the germanium diode, and the closely allied transistor, has brought new hope to computer designers and opened up quite new vistas before them. A transistor which is no larger than a pea can perform many of the functions of a valve; it has no cathode, and it will work efficiently at very low power levels; computing circuits have already been built from transistors, and have been made to work satisfactorily when they were using only a few per cent of the power which would have been needed in an equivalent valve circuit. The manufacturers already claim that the lives of their transistors are as long as those of the better available valves. In order to exploit transistors properly it will be necessary to develop quite new methods of construction, such as, perhaps, printed circuits, and tiny new resistances and so on; otherwise the components and even the interconnecting wiring will be far bigger than the little transistors themselves. Many other new components have been announced in recent years; for example the static magnetic flip-flops and magnetic memories which may ultimately replace the cathode-ray-tube stores and mercury delay lines which take up so much space in existing computers.

If a computer is made very small it becomes increasingly difficult both to build and service it; except in certain special cases there is little to be gained by reducing the bulk of a machine beyond a certain point; nevertheless once ninety per cent of the valves in a machine can be replaced by other components which are both smaller and cooler we can well imagine that the computer of the future will be smaller, more reliable, more compact, and that it will use much less power than what we shall come to regard as the primitive and rather temperamental devices of today.

Digital computers are passing through a stage in their development that radar reached in 1939. The day of the laboratory miracle is long past: the machines work, the first production models are in service, they are vitally important, they are all being operated by highly skilled people; some of their designers are becoming complacent, but the machines have still a long way to go. If the aspirations of their users are not to be disappointed it will once again be necessary to "give them the third best to be going on with; the second best comes too late, the best never comes."* During the war it was necessary to set up operational research groups in order to ensure that complicated machines could be successfully manipulated by unskilled men. We have to train more operators, we may have to redesign the machines so that ordinary mortals can use them more easily, but we have already found that non-graduate girls can be as skilful as the W.A.A.F. radar operators were during the war. We may have to make special computers for special purposes, and we shall have to set up some complicated organizations if we are to be sure that the machines are properly used on sensible projects.

Lest the reader should conclude too hastily that computers will develop with the explosive speed of radar during the war, and that he can profitably wait for a few years before he decides to use one, we must point out firstly that the amount of money spent on them so far is probably only one per cent of that spent on radar, so that progress will be relatively slow; secondly, that current designs may be serviceable for many years to come (the old Home Chain which had been installed by 1939 gave warning of the German V2 weapons at the end of the war); and thirdly that Sir Winston Churchill's aphorism that "New devices are often most effective when they are first introduced and therefore least efficient" may be as true of the weapons of peace as it is of those of war.

The machines which have been built so far are by no means perfect, nevertheless they are very powerful and useful instruments. Better machines will doubtless be built in the future, but for the moment it is above all important to gain experience by using the machines which exist. It would be tragic if those who need them now were to wait, and allow a possible better to be the enemy of the existing good.

* Watson-Watt, *The Story of Radar.*

Chapter 5

PROGRAMMING FOR HIGH-SPEED DIGITAL CALCULATING MACHINES

The four rules of arithmetic may be regarded as the complete equipment of a mathematician—JAMES CLERK MAXWELL

THE FUNDAMENTAL OPERATIONS which an all-purpose machine is required to perform are few in number. In fact, the most complicated mathematical processes may be readily described in terms of the following operations* only—

(*a*) Add the number in memory location n into the arithmetic register.

(*b*) Subtract the number in memory location n from the arithmetic register.

(*c*) Multiply the number in the arithmetic register by 2^{-n} ($n > 0$). This is purely a shifting operation, the alternative (i.e. 2^{+n}) being performed by means of (*a*).

(*d*) "Input" information from an input medium (e.g. teleprinter tape).

(*e*) "Output" information in the required form.

(*f*) Transfer control; this instruction discriminates on the sign of the number in the arithmetic register.

(*g*) Transfer contents of the arithmetic register to a specified storage location.

With these seven instructions, routines which enable a machine to execute more sophisticated operations may be devised and stored —even multiplication and division can be handled in this way. When the frequent use of such orders as multiplication justify it, they can be actually built in as "hardware": such a course, however, reduces the reliability of the machine and, especially when speed is not at a premium, is only justified in a few cases.

THE TECHNIQUE OF CODING

Once the list of fundamental instructions has been fixed, the next problem to be faced by the machine user is that the process he

* This is not intended to be a minimum list—it is a convenient working list for, say, a computer doing some sorts of commercial calculations.

wishes to programme must be broken down into a detailed form acceptable to the machine—i.e. into a set of elementary operations, each one of which corresponds to a machine instruction. To put it another way, we may say that the process must be expressed in the basic machine "vocabulary."

Perhaps the best way of describing how this is done is to examine the details of the lists of instructions (subroutines) used for input and output. These subroutines illustrate many of the "tricks" open to a programmer and at the same time describe processes which are apt to mystify the neophyte. The actual instructions employed are typical of those available in most computers today.

Input and Output

When a problem has been coded, the detailed instructions are, for example, punched on a teleprinter tape by means of a teleprinter punch and a modified teleprinter code: this description will be given in terms of this particular form of input medium, which is the one in most common use at the moment. Facilities for editing such a tape are essential, not only for making corrections but also for combining various subroutines to produce a final tape.

The resultant tape is then fed into the machine input unit (see Chapter 6), which takes in one row of holes at a time and performs suitable operations on the information which it "senses" from the presence or absence of holes in the five possible positions. These operations are dictated either by instructions already taken into the store from the tape, or by a set of *initial instructions*—instructions retained permanently in the machine. These latter are put into the store at the commencement of operation and enable the machine to build up orders from information taken from the tape; thus the machine "tells itself" how to take in and build up instructions. Input of numbers is carried out in accordance with instructions taken in this way.

Output may be accomplished, for example, by means of an ordinary teleprinter. In this case, the five type-setting levers are set up usually to correspond to the five most significant digits in any given storage location, and printing is carried out, 70 digits to a line, on a normal teleprinter "page." All printing instructions must be included in output routines, including figure shift (or letter shift if required), carriage return, etc. In this way, any layout of which the teleprinter is capable may be obtained.

Now that we know how the machine can take in information from the outside world, we can discuss how, in the typical case of information in the form of holes in a teleprinter tape, this information is built up by the machine into instructions or numbers, with the help of an appropriate subroutine. Let us take the case of input instructions first of all.

Suppose, in a simple case, instructions are available in groups, each four rows of holes in length. As each one of these rows may contain up to five holes, it may represent one of 2^5 (i.e. 32) different code combinations. A simple subroutine arranged to take in this information and to place successive instructions in consecutive storage locations might read something like this (it is assumed that, in this case, an instruction is available which will take information from the tape reader and add it into the accumulator)—

Instruction 1. Place the number -4 in a position where it can be used as a counter.

Instruction 2. Clear the accumulator.

Instruction 3. Multiply the number in the accumulator by 2^5 (this is merely a shift operation).

Instruction 4. Take in the next available row of holes on the tape and add the information contained in it to the five least significant digits of the accumulator.

Instruction 5. Add one to the counter; if it is still negative, return to Instruction 3; if positive or zero, proceed to the next instruction.

Instruction 6. Place the contents of the accumulator in a pre-assigned storage location.

Instruction 7. Increase the address of the storage location in Instruction 6 by one.

Instruction 8. Return control to Instruction 1.

There are two things to be noticed about this list of instructions. In the first place, special provision must be made for terminating the process: that is to say, the list of instructions as it stands will cause the machine to continue to take in tape, and does not provide for any means of changing to another programme. This difficulty may be overcome by arranging for special *indicator letters* which the machine can detect, and which will cause a transfer of control to another set of instructions. Thus, supposing the combination corresponding to 11111, i.e. 31, (which is not used for any other purpose) is selected to represent: "this is the end of the input of a series of instructions; change control to the instructions contained in storage location No. —," then we can test for 11111 every time we put in a new row of holes. Our list of instructions may now read—

1. Place the number — 4 in a position where it can be used as a counter.

2. Clear the accumulator.

3. Take the information from the next row of holes and transfer it to the accumulator.

4. Subtract 11111.

5. If the answer is positive or zero, transfer control to the instruction in storage location No. —.

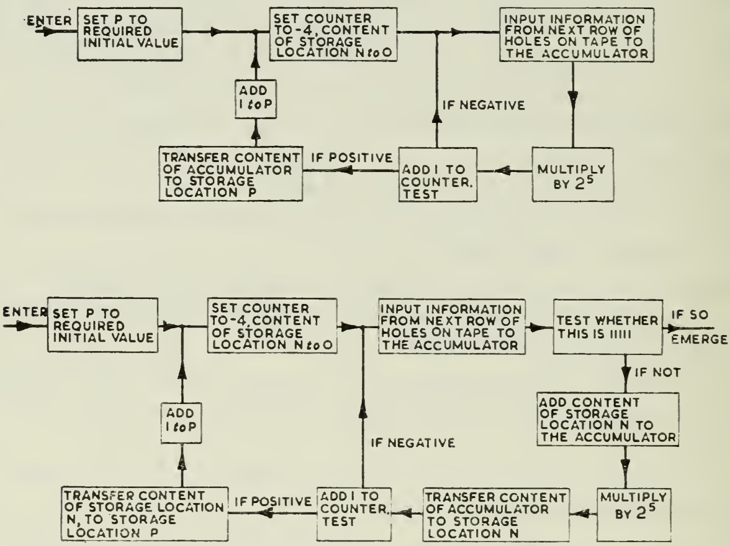


FIG. 5/1. Flow diagrams for input routines

6. If the result is negative, add 11111.

7. Add the contents of storage location $N \times 2^5$ to the accumulator (storage location N will contain the successive forms of the instruction we are building up as they appear at the various stages of this input routine).

8. Transfer the contents of the accumulator to the storage location N.

9. Add one to the counter; if positive or zero, continue; if negative, return control to Instruction 3.

10. Transfer the content of storage location N (which is now the required instruction in its final form) to its correct final storage location.

11. Increase the address in 10 by one.

12. Return control to Instruction 1.

This set of instructions and the previous set are represented in flow-diagram form in Fig. 5/1.

The second point is that an instruction of the type we have described would have its numerical portion already in binary form; no provision has been made in this programme for conversion to binary form. There is no basic difficulty in including such a conversion as part of the input routine, however, and we can arrange for the machine to distinguish between function letters and numerical information in much the same way as it detected the terminating symbol mentioned in the previous paragraph.

The second type of input operation which is most frequently required is the input of numerical information in decimal form. Perhaps we can best describe this operation by detailing a specific routine designed to put an arbitrary number of positive integers into successive storage locations. Let us suppose that the digits 0 to 9 are represented on teleprinter tape in their binary form, and that we select the combination 01010 (i.e. that binary form corresponding to the decimal number 10) to mean, "the number preceding this row of holes is now complete." The fact that the preceding number is the last to be put in will be represented by the combination 01010 followed by 01011 (that is, the decimal number 11 in binary form): the machine can then be made to proceed automatically to another subroutine.

Thus a typical set of instructions might read as follows—

1. Place the information contained in the next row of holes in the five least significant digits of the accumulator.

2. Subtract 01010.

3. If negative, continue; if positive or zero transfer control to Instruction 10.

4. Add 01010. (We have determined at this point that the information we have just taken from the tape is the digit of a number.)

5. Add the contents of the accumulator to the contents of storage location N; the sum will appear in the accumulator.

6. Transfer the contents of the accumulator to the storage location $N + 1$.

7. Multiply the contents of storage location $N + 1$ by 01010, and place the result in the accumulator. Note that 01010 can be placed in the multiplier register at the beginning of the operation, and it will remain there throughout the subroutine.

8. Transfer the contents of the accumulator to storage location N.

9. Return control to Instruction 1.

10. According to whether our original combination was 01010 or 01011 we now have either 00000 or 00001 in digit positions 0–4 of the accumulator. We must differentiate between these possibilities by subtracting 1.

11. If the result is negative, continue; if zero, transfer control to Instruction No. — (i.e. input of the list of integers has now been completed).

12. We have at this stage completed the input of the integer which is now standing in converted form in storage location $N + 1$. Transfer the contents of storage location $N + 1$ to the accumulator.

13. Transfer the contents of the accumulator to the final storage location which we require them to occupy.

14. Increase the address in Instruction 13 by 1.

15. Clear storage location N.

16. Return control to Instruction No. 1.

The instructions used above are typical of those available in most digital computers. Their exact form will determine the details of the input programme in individual cases: the routine in outline will be something like that described. For the benefit of the reader who is not convinced that the mathematical process described will give him the right answer, the logic of the process becomes obvious when we recall that the number 205 may be written in the form

$$[(2 \times 10 + 0) \times 10 + 5]$$

To complete the picture, we shall describe how information which is in the machine in the form described above would be printed out by teleprinter. A teleprinter operates by being fed with groups of five digits, which it interprets as teleprinter code symbols. For the purposes of this exposition, we can assume that the teleprinter is already on figure shift, and that such ancillary instructions as line feed and carriage return are taken care of separately. Suppose we know that all the numbers which we require to print out are positive and lie between 0 and 1,000; then we would proceed something like this—

Instruction 1. Place -3 (as a counter) in storage location T.

Instruction 2. Take the content of the accumulator (which we wish to print out) and subtract from it 1100100 (i.e. the decimal number 100 in binary form).

Instruction 3. If result is negative, transfer control to Instruction 6.

Instruction 4. If the result is positive or zero, add 1 to a counter in storage location S.

Instruction 5. Transfer control to Instruction 2.

Instruction 6. The counter will now contain the first decimal digit which we wish to print out in the least significant portion of storage location S. If our teleprinter has been arranged to accept information from the five least significant digits of the storage location, then this instruction will simply read: "Output the five least significant digits of storage location S via the teleprinter, and leave location S clear."

Instruction 7. Add 1100100 to the accumulator.

Instruction 8. Transfer content of accumulator to storage location Q.

Instruction 9. Multiply the contents of storage location Q by 01010 (i.e. the decimal number 10 in binary form).

Instruction 10. Add 1 to the counter in location T; if positive or zero, continue to the next instruction, and if negative return to Instruction 2.

Instruction 11. We have now completed the output of our three-digit number; this instruction will transfer control to the next subroutine which we desire to enter.

The verification that this process will give the correct result is left as an exercise for the reader.

The Technique of Programming

Now that we have examined the way in which the individual instructions are handled, we can discuss in more general terms the procedure to be adopted when we are faced with an actual mathematical problem.

There are three phases involved in breaking down such a problem to a stage at which it can be punched on to the tape—

(*a*) *Macro-programming.* The resolution of a programme into separate large-scale processes. This, and (*b*) below, are best done by means of block diagrams giving a picture of the general flow of computations. Various notations have been introduced from time to time, but programmers often prefer to use their own conventions. Suppose the problem is to solve numerically the equation $\tan ax = x$ for various values of a over a certain range. The macro-programme would perhaps be represented diagrammatically as in Fig. 5/2.

This rough schedule is useful for obtaining an overall picture. The second stage can now be approached.

(*b*) *Micro-programming.* In this example only the Newton-Raphson process need be detailed further before the coding proceeds. This process, it may be recalled, can be described in the following words—

If x_1 is an approximate solution of the equation $f(x) = 0$, then $x_1 + h$ is a better solution, where $h = -f(x_1)/f'(x_1)$.

Here $f(x) = \tan ax - x$, and $f'(x) = a \sec^2 ax - 1$.

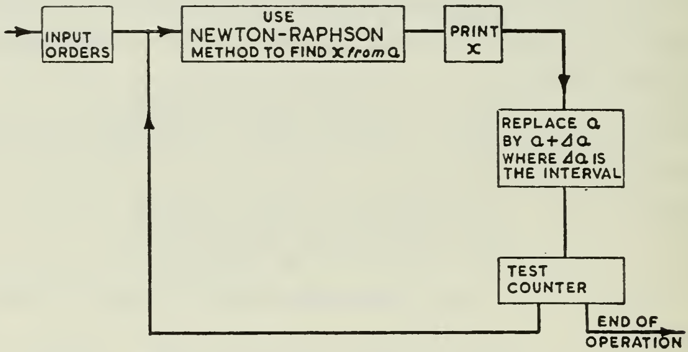


FIG. 5/2. Example of a macro-programme

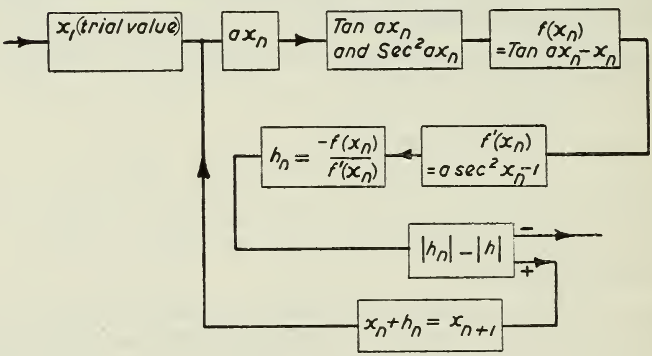


FIG. 5/3. Example of a micro-programme

Thus we require a routine for finding tan *ax* as part of the programme; this also yields $\sec^2 ax$, as $\sec^2 ax = \tan^2 ax + 1$. A sample micro-programme would then be as shown in Fig. 5/3.

Two processes requiring extra programming in most machines are included in the above, i.e. forming tan *ax*, and the division involved in forming *h*. As both of these processes are likely to be required often, they are usually coded once and for all and exist, e.g. as tapes in a tape-library from which they can be extracted as required.

(*c*) *Coding.* Each of the individual blocks depicted above must now be broken down into detailed instructions specific to the particular machine for which we are coding. It is this process which is most susceptible to error of a mechanical nature, and precautions are taken wherever possible to ensure that these errors are eradicated before the final programme is presented to the machine. Examples of coding are given in connexion with input and output routines above.

Use of Subroutines

As already mentioned, standard processes are coded and stored as subroutines, in a library, or perhaps on a magnetic drum. With some machines, these subroutines are so arranged that they are modified at the input stage, both so that they can be used in any range of storage locations, and also for any other parameters required; with other machines, the technique of using subroutines in standard storage locations is followed. These subroutines are usually constructed to be as flexible as possible—one subroutine, for example, may be used for the solution of any number of linear simultaneous equations (see page 396).

Library Facilities

Over a period of time a collection of flexible subroutines which are economical in time and storage requirements becomes available. Whole processes, once constructed, are useful for all time with a given machine, and their generality should be such that, by modifying a few parameters, they can be applied to a range of similar problems, store size permitting.

Routines and subroutines of this type represent the investment of a considerable number of man-hours. Coding time is usually only a fraction of the total time involved in getting the programme to work—such is the frailty of mankind. Moreover, many of the short cuts so dear to hand computers may so complicate an automatic programme that their inclusion is seldom worth while; the machine has as much "intelligence" as you give it, and many errors are made in the giving. Therefore, when a decision is being made as to whether a problem should be attacked by hand or automatic computation—or by a combination of the two—a library containing programmes for similar problems will weigh as a heavy "pro" for machine work.

Checking Routines

As so much time is involved in locating and correcting coding errors, the best way of carrying out this process is by arranging that the machine should print out sufficient information, while performing a programme, to enable the programmer to follow the course of the computation. In this way he can readily detect where his programme has gone astray and make appropriate corrections.

Such a process is carried out by means of standard checking subroutines. These, for example, may print out in succession the instruction letters to each instruction in turn as it is obeyed, or perhaps the contents of the accumulator at certain pre-arranged points in the calculation.

An essential feature of this technique is that it becomes possible for a programmer to make corrections to his programme away from the machine; the alternative of having the machine proceed, instruction by instruction, under the supervision of the operator, necessarily consumes more machine time.

Programming and Reliability of Machines*

Operators are often forced to use machines which are not 100 per cent reliable—a fact which need not necessarily make them useless for computation. A simple procedure which is often adopted in these circumstances is similar to that used in hand computation, i.e. a mathematical checking procedure is carried out in parallel with the actual computation. If this check is examined at pre-arranged points of the computation and indicates that an error has occurred in the previous stage, the machine can be made to repeat this stage again and again until the check is satisfied.

This procedure, it will be noted, is equivalent to the acceptance of a slightly lower degree of reliability at a cost of increased time of computation; it is quite normal in any manual computation, and is fast becoming accepted as routine with machine calculations of any length. The number of check points, of course, will be determined by the expectation of reliability of the machine in question, and the nature of the calculations.

Optimum Programming

The simple coding procedure described above does not make the best possible use of some types of machine—to do this the technique of *optimum programming* must be introduced.

* See also Chapter 4.

With any storage medium which does not permit immediate access, a considerable time may be wasted waiting for the required information to become available. Thus, if information is stored around a magnetic drum, access being by means of a single magnetic head, we will have to wait for a complete revolution of the drum if we wish to recover information which has just passed our reading head.*

Suppose we alter the order in which instructions are obeyed, no longer carrying them out in the sequence in which they are stored. If we arrange that when we have finished the operations associated with a given instruction, the next instruction which becomes available at the reading head is the one which we require, we have a method of speeding up the computation considerably. We may, in fact, select the instruction which next becomes available after the present instruction has been completed—or alternatively we can select the next instruction at any time during the execution of the present one.

The first difficulty which arises when we come to apply this technique is that we may not always be able to find a free position just where we want it, even if we are prepared to select from any track. To get over this, we should make the system sufficiently flexible to enable instructions to be selected from any position if required.

The second difficulty, which is far more fundamental, is that which arises if we are operating on a series of numbers. If these numbers are not immediately available, our system will be upset; and so, when this optimum-programming technique is employed, a number of quick-access stores is usually employed. Another way of minimizing the difficulty, employed in the A.C.E. pilot model, is to make provision for the carrying out of the same arithmetic operation (multiplication and division are not included in this case) on two complete series of numbers available in succession from two different sources, the results being placed in turn in a given sequence of final locations. The first arrangement will make a machine more useful for solving problems where only a small amount of numerical data is involved, and the second for some types of manipulation of sequences of numbers.

The use of this technique complicates programming somewhat; however, as most computing time is usually taken up executing

* The terminology used here is applicable to magnetic-drum storage; the same arguments apply, of course, to other media which make information available in serial form, such as acoustic delay lines.

standard subroutines, we can afford to forget about maximum speeds in a main routine, and to depend on fast library subroutines for our increased speed. The penalty incurred in machine time will often be more than counteracted by the saving in programming time, especially for a newcomer.

Interpretative Techniques

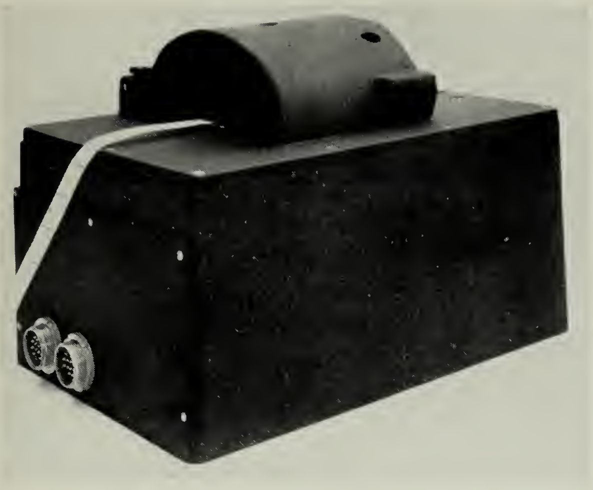
The technique of starting with the simplest possible type of routine—that necessary to assemble other instructions—and of using it to build up a library of subroutines of increasing complexity enables us to make a very considerable advance in ease of coding. It becomes possible to design a code which is as straightforward as we require and which may have very little resemblance to the way in which information is held in the store. The interpretation of this input code can either be done by the machine as instructions are being put in initially, or can be carried out whenever the machine comes to the subroutine which has been coded in the interpretative code.

The way in which this interpretation is carried out is simple enough. We merely use the digit patterns resulting from our special code to operate a large number of *transfer-of-control* instructions which take us in succession to various "dictionary" entries—sequences of orders in the machine's own language.

Automatic Coding

It has been found possible to use the Manchester machine to convert programmes written in a notation very similar to simple algebraic notation into its own code. The programme, in this simple form, is fed into the machine under the control of a special input routine which makes the translation into the code of the machine. The notation has been designed to be as near as possible to the usual notation of algebra, so that the construction and checking of programmes is made easy; this lessens the difficulty that is found in constructing large and complicated programmes.

The notation consists of two parts; one for the description of the numerical calculations, and the other for the description of the organization of the calculations into a completely automatic process. The description of the numerical calculation is in the form of algebraic "equations" using the simple basic operations of addition, subtraction, and multiplication, as in the following equation which



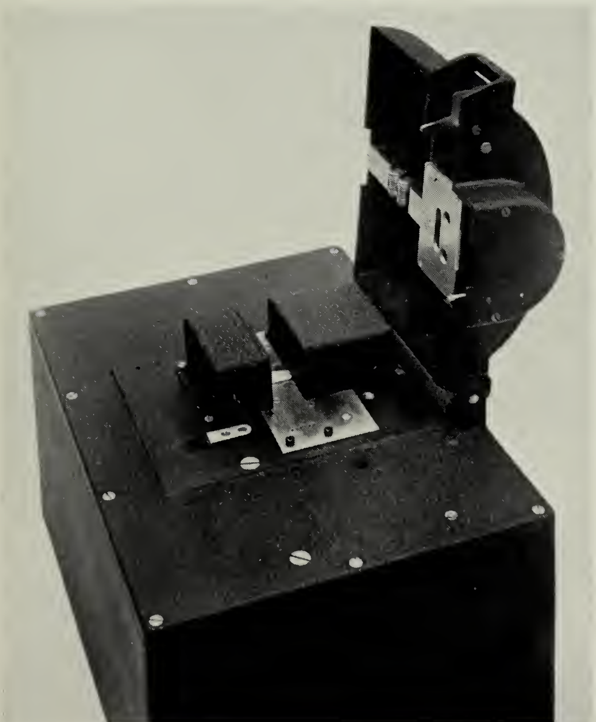


PLATE V. THE PHOTO-ELECTRIC TAPE-READER OF THE MANCHESTER MACHINE

Above. Showing a tape being fed in.

Below. Showing tape guide and sensing holes.

describes the addition of several numbers contained in storage locations for which the code letters are x, y, z and so on. It is—

$$+x +y +z +a +b \rightarrow c$$

The result of the sum is to be found in the storage location c. These locations are not fixed, but may be defined according to need. This type of notation is sufficient for the description of most numerical calculations.

For the description of the organization, which forms a considerable part of any programme, instructions take the form of words (in English) which, when interpreted by the machine, cause the correct instructions in the machine's code to be synthesized. Thus if a subroutine is required during the calculation, for printing the results or the calculation of auxiliary functions, it is sufficient to write the word *subroutine* followed by a number describing which subroutine is meant. By an extension of this technique it would be possible to call for the particular subroutine by name (e.g. *cosine* for a subroutine for calculating cosines). This has not yet been done as the gain in convenience would be too small to warrant the trouble.

In the system now in use, English words are used to specify transfers of control, counting and many of the more common programme tricks. The total number of descriptive words is 13, and this has been found adequate for most purposes.

Programming with such a system for making the machine do its own coding does not lead to the most economical or fastest programmes, but the loss in "efficiency" is not more than about 10 per cent and is a small price for the convenience that results.

REFERENCES

1. Wilkes, M. V. *J. Sci. Instr.*, **26,** (1949) 217.
2. Wilkes, M. V., Wheeler, D. J., and Gill, S. *The Preparation of Programs for an Electronic Digital Computer, with Special Reference to the E.D.S.A.C. and the use of a Library of Subroutines.* Addison-Wesley Press Inc. (Cambridge, Mass. 1951).

PART TWO

ELECTRONIC COMPUTING MACHINES IN BRITAIN AND AMERICA

Chapter 6

THE UNIVERSITY OF MANCHESTER COMPUTING MACHINE

THE COMPUTING MACHINE which is now operating at the University of Manchester (M.A.D.M.) represents the culmination of a research project of several years' standing. It seems appropriate to outline the various steps in the development of this project, since these have given the final machine its major characteristics.

The story begins at the Telecommunications Research Establishment at the end of the war. The development of Radar had produced many new techniques, and it was natural to seek other fields for their application. The National Physical Laboratory were already interested in computing machines, and the A.C.E. was in its early stages.

In the United States, experiments were being made in an endeavour to store radar traces on a cathode ray tube. Another American source reported that the observed phenomena were quite unsuitable as the basis of a computing-machine store. The reason given was that the "memory" was too transient, and thus the record could be read only a very few times before it faded away. In spite of this unfavourable report, the application of cathode-ray-tube storage to computing machines was considered to be so attractive, if it could be achieved, that a research project with this as its aim was started late in 1946. Looking back, it is amazing how long it took to realize the fact that if one can read a record once, then that is entirely sufficient for storage, provided that what is read can be immediately rewritten in its original position. Experiments with a few binary digits proved that this method of storage by "regeneration" was possible on the screen of a cathode ray tube, and justified further research.

In January, 1947, the project moved to Manchester, but continued to have, and fortunately still has, the active support of the Telecommunications Research Establishment.

The original store used the *anticipation pulse system*, but since then many other configurations have been tried.[1] The one in favour at the moment is the defocus-focus system (see pages 62 and 148). The successful operation of a single cathode-ray-tube store, containing 1,024 digits, for a period of hours during the autumn of 1947 finally set the main line of the design of the machine, since from then on it was clear that it would use cathode-ray-tube storage.

A Prototype Machine

The next development was to build a small prototype machine.(2) This machine, a schematic diagram of which is shown in Fig. 6/1 and a photograph in Plate VII used one cathode ray tube as a store,

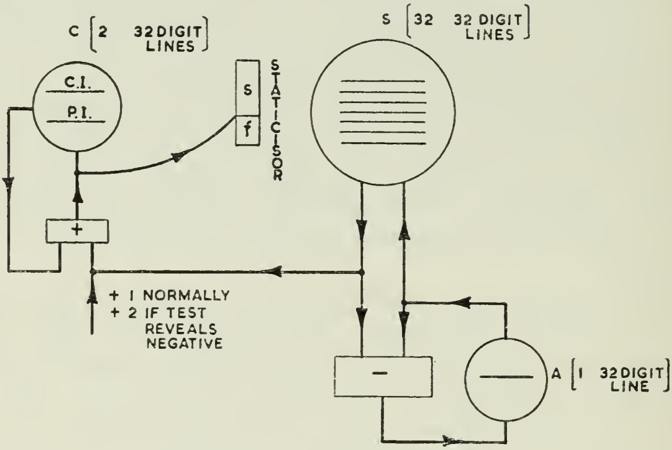


Fig. 6/1. The prototype machine

S, to hold 32 words, each of 32 digits. It operated serially on the binary digits of a number, represented negative numbers by complements, and used the single address code, an instruction being of the form (*s*, *f*), where *s* specified a store address, and *f* the function to be performed. It set the tone for subsequent Manchester machines in all these respects. A second single-word C.R.T. store was used as the accumulator *A*, all calculations being performed by transfers of numbers between store and accumulator, whilst a third C.R.T. store *C* held the control number recording the number of the instruction being performed. Since an instruction must be read from the store *S*, and must then itself control the store selection mechanism via the staticisor, each instruction must be held in temporary storage as it is read from *S*. A second line on the control tube *C* is a highly convenient place for this temporary storage, since the control number and the temporarily stored instruction are then both fed to the staticisor from the single source, *C*. The two lines of *C* are called the C.I. and P.I. lines, the initials referring to *control instruction* and *present instruction* respectively. Taking the duration of the sweep of one store line, i.e. the number length, to be one "beat," and the time taken to find, read, and obey an instruction as being one "bar," this machine operated with four beats to the bar, as shown in Fig. 6/2 (*a*). In the first beat, S_1, the control instruction

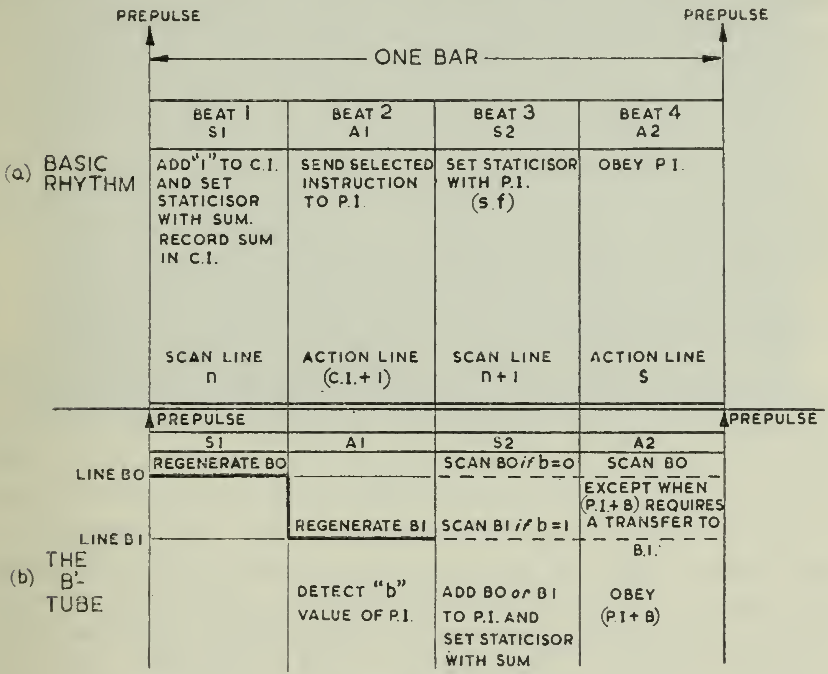


Fig. 6/2

was read from C.I., increased by unity to cause the instructions of a programme to be obeyed sequentially, rewritten in C.I., and fed to the staticisor. Meanwhile the store *S* regenerated some line, say *n*. In the second beat, A_1, the staticisor selected a line of the store *S* and fed the appropriate present instruction to the P.I. line. In the third beat, S_2, the present instruction was read from P.I. and fed to the staticisor, while the store *S* regenerated line $n + 1$. In the final beat, A_2, the staticisor controlled the store, and performed the operation specified by the present instruction. It can be seen that during beats 1 and 2 the machine was under the control of C.I., while during beats 3 and 4 the machine operated in a similar manner, but was under the control of P.I. During the scan beats S_1 and S_2, successive lines of S, namely n and $n + 1$, were regenerated, and the staticisor was set whilst the store was in this passive state; but during the action beats A_1 and A_2, the store was in an active state, the line scanned being determined by C.I. or P.I. via the staticisor. Basically this rhythm of four beats in a bar, which was initiated by a "pre-pulse," has remained unchanged in later machines, though the addition of facilities of duration greater than one beat has entailed some modification after the beat A_2. This prototype machine could subtract from the accumulator, write negatively to the accumulator, write from the accumulator to the store, write to control, and add to control. It also had a test facility on the sign of the content of the accumulator, which, if the sign was negative, permitted conditional skipping of an instruction by adding 2 to C.I. (see Fig. 6/1). Clearly this machine had the bare minimum of facilities, but it operated successfully on small programmes in June, 1948.

A Magnetic Store

Up to this point the fact that a full-scale machine with a storage capacity of about 4,000 words would call for over 100 C.R.T. stores had been recognized, disliked, and forgotten. Now this problem had to be faced. Magnetic stores had, of course, been propounded for other machines in a variety of forms; wire, or tape, was proposed for use in conjunction with other stores, and drums suggested as main stores. Wire and tape appeared to have two disadvantages. First, time would be wasted in searching for information along the medium, and secondly, some means of identifying individual digits, or blocks of digits, on the medium would be necessary. Since one objective of machine design was economy and a resulting reduction in the necessary number of C.R.T. stores, it was obvious that the best solution would be one that permitted the reloading of a C.R.T. store in the minimum time. The absolute minimum time is the time taken to sweep through all the digits on a C.R.T. at the digit repetition rate. To achieve this it is necessary to make digits emerge from the magnetic store in exact synchronism with the sweeping of spots on the C.R.T. store. This was achieved by using a magnetic prum rotated under the control of a servo system[(3)] that related drum position accurately to the position of a spot sweeping out the dattern on the C.R.T. This involved controlling the position of a drum rotating at about 2,000 r.p.m. to an accuracy of about plus or minus one hundredth of a degree, but was found to be fairly simple,

Plate VI. A Typical Stored Pattern on a Cathode-ray-tube Screen

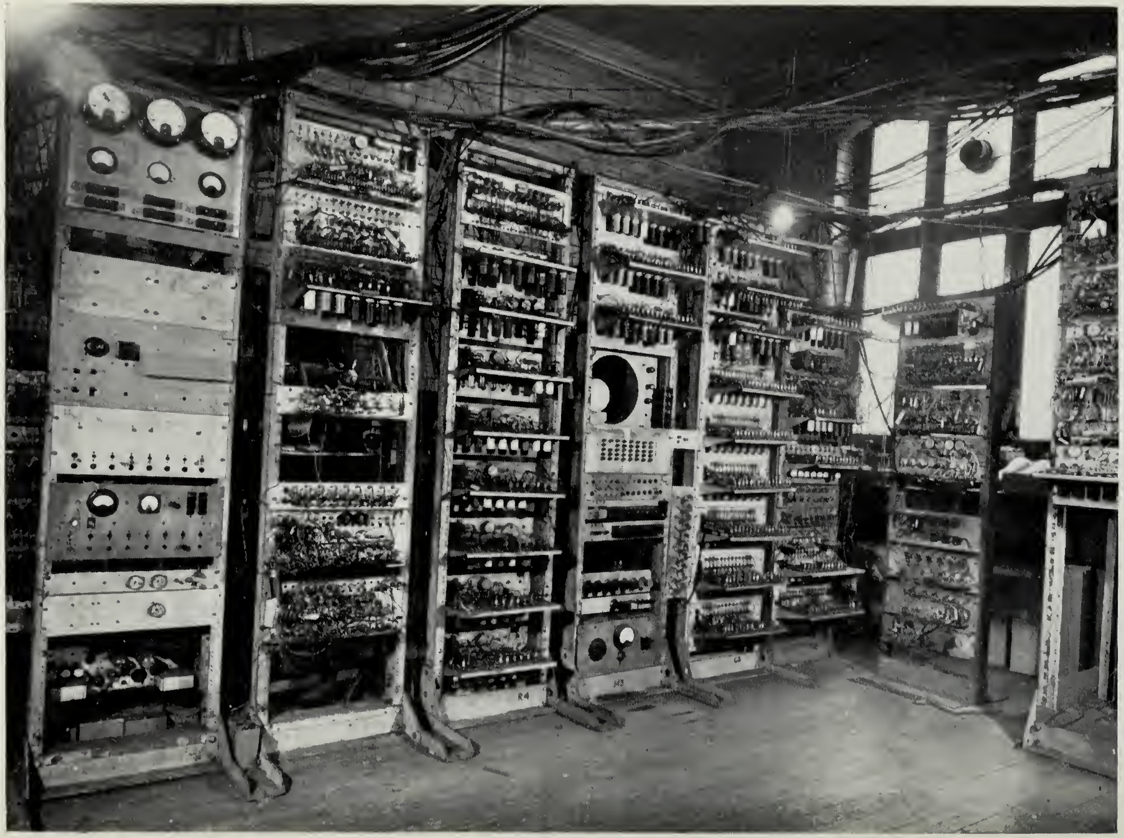


Plate VII. The First Manchester University Computer granted good mechanical design of the drum. This combination of magnetic and C.R.T. storage permits indefinite extension of storage capacity, by using a series of parallel tracks along the length of one drum, and by the addition of further drums. Rapid selection of any track is made possible by providing each with its own magnetic head.[4]

With this arrangement the magnetic store becomes the major storage element, the C.R.T. store being used as a sort of "smoother" to cut down the average rate of reference to the magnetic store to a figure appropriate to the 30 msec access time of that store. One cannot

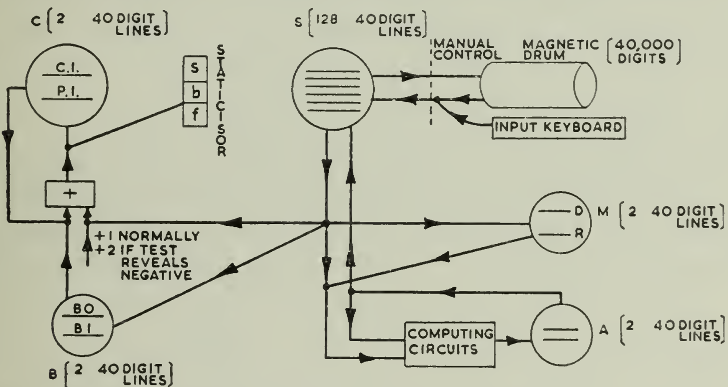


FIG. 6/3. The first 1949 machine

be specific, without considerable programming experience, about the amount of "smoothing" that will result, but clearly with a bigger C.R.T. store there need be less frequent reference to the magnetic store. Conversely more rapid reloading from the magnetics permits the use of fewer C.R.Ts. There is room here for a nice economic balance, since longer access time makes possible a more economic magnetic store but, for lack of experience, the optimum balance is not yet known.

The synchronized magnetic drum was the second major controlling factor in the development of the machines at Manchester and continues to provide the bulk of the storage in the present model.

An Improved Machine

Two further machines incorporating the magnetic store were built during 1949. The first,[5] shown schematically in Fig. 6/3, was an extension of the prototype machine with more C.R.T. storage, namely 128 words of 40 digits, and more facilities, including an electronic multiplier.[6] The accumulator A had associated computing circuits for addition, subtraction and three logical operations. Multiplication involved the use of the storage tube M. The multiplicand was stored on the D line and remained there for possible further use after a product was taken; and the multiplier was stored on the R line. Products were added into the accumulator automatically, appropriate account being taken of the signs of D and R. Two 40 digit lines were provided for the accumulator, so that the whole of an 80 digit product of two 40 digit numbers was retained, and any "rounding off" required was entirely under the control of the programme. Computation using the accumulator was performed modulo 2^{80} by extending forty-digit numbers from the store S by 40 copies of their most significant digit.

The magnetic store of this machine had a capacity of 40,960 digits, but loading from the magnetic to the C.R.T. store, and vice versa, could only be achieved manually. Nevertheless, during the summer of 1949, useful work was done on the machine. In particular the Mersenne numbers $2^p - 1$ were tested for primality by the Lucas test. By this means the primality of those Mersenne numbers, already known to be prime from other calculations, were checked. The values of p greater than 257 and less than 354 were also tested and the corresponding Mersenne numbers found to be composite.

A new feature, which was introduced by this machine, and which in an extended form is a part of the latest machine, was the B-tube. This was a storage tube having two lines B_0 and B_1, either of which could be swept at will under the control of a single b digit in the instruction, which now took the form (s, b, f). The output of the B-tube was added to P.I. during S_2, before P.I. was used to control the staticisor (see Fig. 6/2 (b)). Thus instructions, and in particular their address section, could be modified in their effect without being modified in their stored form. In general, line B_0 was kept empty, so that instructions were used unmodified for $b = 0$, but were modified for $b = 1$. This device was primarily intended as a convenient means of shifting the effect of a whole block of instructions by a constant amount, whilst leaving others that were not B-modified unaffected. Since then this tube has found many other similar applications.

Instructions could, of course, still be modified by the more normal processes using the accumulator, but this was very often inconvenient and wasteful of time and storage space. The numbers contained on

B_0 and B_1 were sent from S to the B-tube by the standard transfer process, and arrived in the B-tube during the A_2 beat.

A Large-scale Machine

The second 1949 machine was an extension of the first, the most important development being that by which the magnetic store was brought under the control of instructions from the machine, reference to it being made fully automatic. The design of this facility was influenced by a desire to permit reference, not only to the magnetic store, but also to other possible stores, input and output systems, and so on, without a re-coding of the machine being necessary as these further facilities were incorporated. It was also thought desirable for simplicity and economy to preserve the normal rhythm of the machine as far as possible. To achieve these results a single function, f_0, was set aside to be used as part of a standard type of instruction (s, b, f_0). In the chosen address s, as part of a programme, a key word was placed, the meaning of f_0 being that the key word was to be set up on a staticisor. This staticisor was called the *magnetic* staticisor and it was set to the key word during the beat A_2, by the normal processes of the machine, as shown in Fig. 6/2. By suppressing the prepulse, normally given after the beat A_2, for the function f_0, the machine was then completely under the control of the key word via the magnetic staticisor. For magnetic transfers the key word took the form (T, E, F) where T stated the number of the track required on the magnetic drum, E the C.R.T. store required, and F the direction of transfer, i.e. whether the content of track T should be placed in the store E, or vice versa. The transfer of information took place, of course, during beats subsequent to A_2, and occupied a known time, so that a prepulse could be given after the transfer was complete, and normal operation resumed.

In previous machines the input to the machine had been directly to the C.R.T. store from a binary keyboard, and output had been by inspection of a C.R.T. monitor. For this machine, input and output routines were developed which enabled teleprinter equipment to be controlled by the machine. The flow of information to and from the machine was via the last five digit positions in the accumulator. The routing of information between these five places, the C.R.T. store, and the magnetic store was an entirely "programmed" procedure. This principle of using a programmed input and output, as distinct from employing special equipment, has been largely adhered to since.

A number of programmes were run on this machine before it was closed down in the summer of 1950. In particular the Riemann hypothesis was investigated and verified for the range $63 < (t/2\pi)^{\frac{1}{2}} < 64$. This chiefly involved calculating the Riemann Zeta function for about a thousand values of t. For this purpose the values of log n and $n^{-\frac{1}{2}}$ were taken from a table, log $(t/2\pi)$ was calculated without reference to a table, and the cosines were obtained by linear interpolation in a table with interval $\pi/128$. The time for each term of

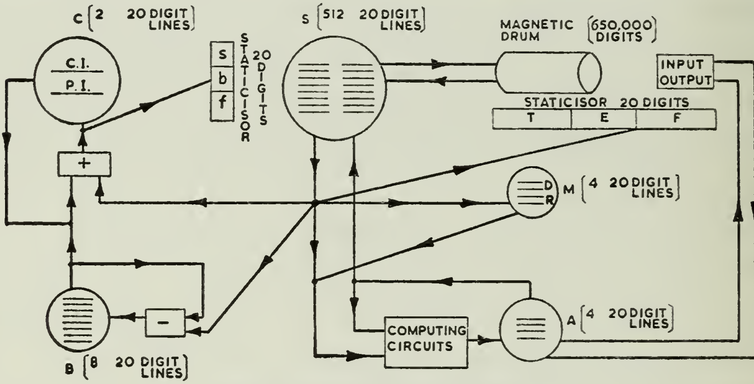


FIG. 6/4. The present machine

the series was about 160 msec. Further, a problem concerned with ray tracing through a lens system was investigated. Most of the rays were skew.

The Present Machine

The machines described in previous paragraphs were entirely concerned with engineering and mathematical experiment. On the basis of experience with these machines, the design of the present machine(7) was laid down in mid 1949 and developed in detail, in close co-operation with Ferranti, Ltd. A statement of the basic design principles, which also serves as a summary of features already discussed, is as follows—

1. The machine (see Fig. 6/4) contains—
   (*a*) A C.R.T. store using the defocus-focus system.
   (*b*) A magnetic store of the synchronized drum type.
   (*c*) A *B*-tube.

2. It has a twenty-digit instruction of the type (s, b, f).

3. It operates serially on the forty binary digits of a number, the digit frequency being 100 kc/s.

PLATE VIII. A GENERAL VIEW OF THE MANCHESTER UNIVERSITY COMPUTER WITHOUT COVERS, WHILE STILL AT THE FERRANTI WORKS

4. It has a rhythm based on that shown in Fig. 6/2.

5. It has an eighty-digit accumulator operating in conjunction with an electronic multiplier.

6. It uses the key-word method for referring to the magnetic store and input and output.

7. It uses teleprinter equipment for programmed input and output.

These items were all, of course, included in some form in the final experimental machine, and in describing the present machine it will only be necessary to mention improvements and modifications.

Storage. Experience of the large-scale experimental machine showed that the C.R.T. storage capacity of 5,120 digits was rather too small to permit flexibility of programming, and that the subsidiary storage, on the magnetic drum, of 40,960 digits would be quite inadequate for many problems. The C.R.T. storage capacity has therefore been increased to 10,240 digits, using eight cathode ray tubes. A typical stored pattern on a C.R.T. is shown in Plate VI. The magnetic storage capacity has been increased to 150,000 digits, provision being made for a further increase to a maximum of about 600,000 digits if this is necessary. The magnetic drum (see Figs. 2/25 and 2/26) is 10 in. in diameter, and along its length of 12 in. there is sufficient space to accommodate 256 separate peripheral tracks. Each track holds 2,560 useful digits, the packing density being 100 digits per inch.

Key words, 20 digits in length, cause a transfer of information between the contents of a stated track and a stated pair of cathode ray tubes. Alternatively, if it is more convenient, the transfer of information may be between one half-track and one cathode ray tube. In each case the correctness of transfer can be checked by a single instruction, and, in the case of an incorrect transfer, the transfer can be repeated, or the machine stopped.

Selection of a particular track for reading is by electronic switching, and for writing, by relay switching.

The time taken for the check, and for any transfer in which a track is read, is 36 msec. For a transfer in which information is written into a track, this time is increased to 63 msec to allow time for the relay switching. This discrepancy between the times for reading and writing is not important, because reading is more frequent than writing. This is so because the sub-programmes required during computation are all stored on the magnetic drum.

The Rhythm. To increase the speed of operation of the machine by a factor of between 1·8 and 2, the rhythm has been based on a beat whose length is determined by the twenty-digit instruction, and not by the forty-digit number. This is advantageous because inspection of Fig. 6/2 shows that, of the four beats in the bar, only A_2 is required to be of number length. Further, for operations involving the *B*-tube and modification of the control instruction, numbers of twenty-digit length only are required, so that in these cases the bar is exactly as in Fig. 6/2, but requires only half the period

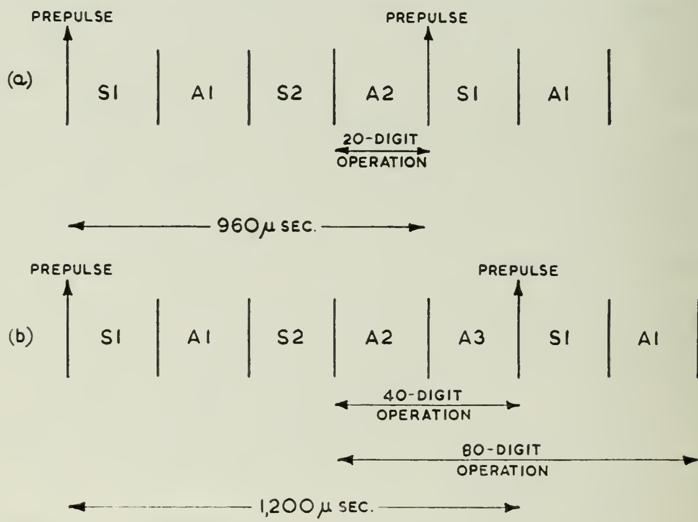


Fig. 6/5. The rhythm of the present machine

of time (Fig. 6/5 (*a*)). For operations requiring forty-digit numbers, as, for example, when the multiplicand is transferred to the multiplier tube, the bar is extended by a further action beat A_3, as shown in Fig. 6/5 (*b*). For those operations using the accumulator, which require an eighty-digit number (a forty-digit number extended by forty copies of the most significant digit), the prepulse is still given after A_3 to initiate the next bar. This is permissible because only the accumulator is involved in completing the eighty-digit operation, and therefore the last two beats required may be the S_1 and A_1 beats of the next bar.

The Accumulator. The facilities for performing addition, subtraction and three logical operations have been retained, together with signed and unsigned multiplication, in which the eighty-digit product of two forty-digit numbers is added to the content of the

PLATE IX. A GENERAL VIEW OF THE MANCHESTER UNIVERSITY COMPUTER AND THE CONTROL DESK, WITH TWO OF THE FERRANTI ENGINEERS

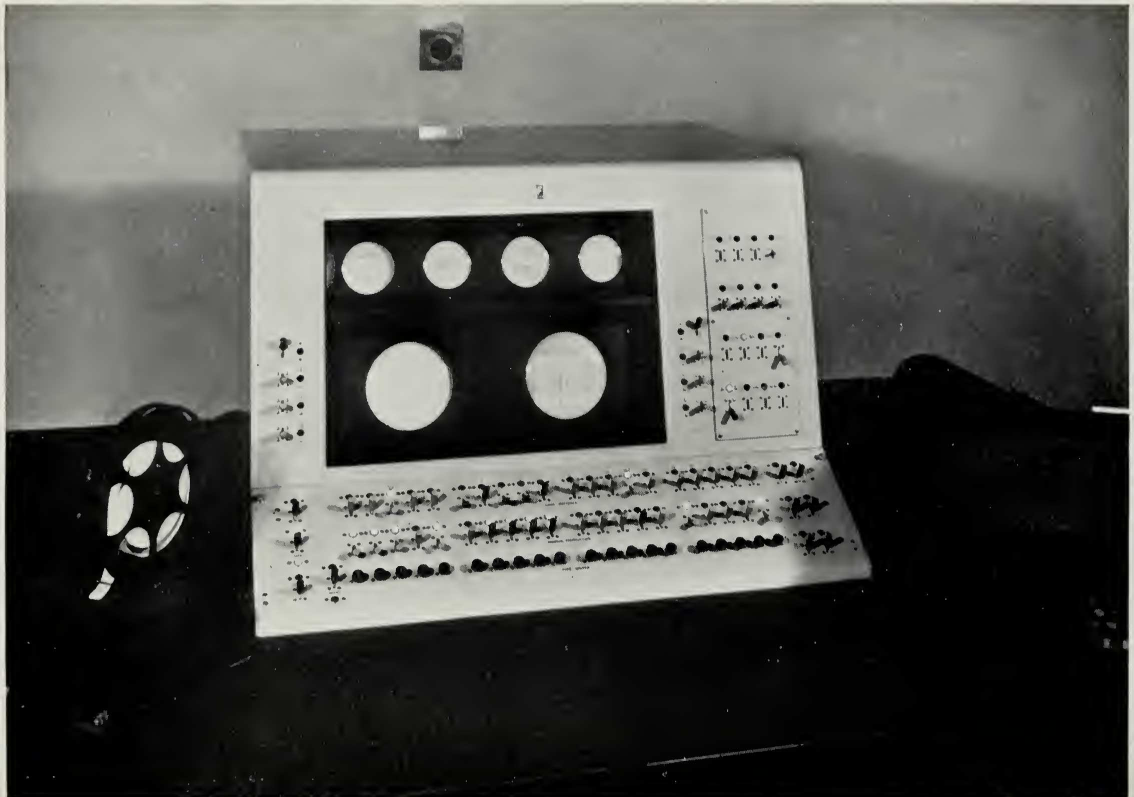


PLATE X. THE CONTROL DESK OF THE MANCHESTER UNIVERSITY COMPUTER, SHOWING THE CONSOLE accumulator. Multiplication in which the product is subtracted from the accumulator is a new facility.

The most important engineering change of the accumulator is in the multiplying circuit. In the experimental machine the multiplication time, which depended on the number of ones in the multiplier, averaged about ten times that for addition. A study of available programmes revealed that, if this multiplying circuit were used in the new machine, about half the computing time of the machine would be spent in multiplication. Consequently, the speed of multiplication has been increased, and, in the new machine, an instruction involving multiplication is selected, interpreted and obeyed in 2·16 msec, as compared with a corresponding time of 1·2 msec for addition, or any other forty-digit number operation. The increase in speed of multiplication is obtained by using a circuit consisting of network of adders and delays, arranged in such a way that the sub-products are added together in one operation instead of sequentially. Although some economy of equipment is achieved by carrying out the multiplication in two parts, controlled respectively by the first and second halves of the multiplier, nevertheless this type of multiplier is expensive and increases the size of the final machine by about 10 per cent (the machine contains 1,600 pentodes and 2,000 diodes). This, however, is judged to be worth while when balanced against the resulting overall increase in speed of operation of the machine.

To simplify the standardization of numbers during the course of computation, an instruction has been provided which produced in the accumulator a statement in binary form of the position of the most significant digit of any selected number in the electronic store.

An additional circuit will generate (in 5·8 msec) a twenty-digit random number as a result of a single instruction. This instruction can be used in calculations concerned with stochastic processes.

The B-tube. This facility has been improved to allow still greater flexibility and economy in programming. Eight twenty-digit storage locations can now be selected by the three *b*-digits of an instruction, and thus any one of eight words may be added to an instruction before it is used.

Further, the *B*-tube has been provided with a subtracting circuit, and a sign-testing circuit. Any line of the tube can therefore be used as a twenty-digit accumulator for simple arithmetical processes. One important example of this is in counting the number of times that a group of instructions has been obeyed, transfer of control being conditional on the sign of a number of the *B*-tube. By this means four instructions are removed from any loop in which counting is required.

The instructions provided enable a number from the C.R.T. store to be written into, or subtracted from, the *B*-tube; or a number in the *B*-tube to be written into the C.R.T. store. The line of the *B*-tube involved in these instructions is stated by the three *b*-digits. To avoid the necessity for eight test circuits—one for each line—the instruction "test the sign of *B*" reads a single circuit, which is recording the sign of the content of that line of the *B*-tube which was last used.

Input and Output. The input-output system is controlled by key words via the magnetic staticisor. The input uses a photo-electric tape reader capable of reading 200 five-digit characters per second. The output is by a mechanical tape-punch and/or teleprinter capable of printing 15 and 6 characters per second respectively.

The speed of input is approaching the maximum for this machine if an input programme is required to organize the input information. However, the speed of output, which is twenty times as slow, can only be justified if, in general, the time of output is smaller than, or comparable with, the input and computing times together. If this proves to be untrue, an alternative scheme for input and output which has been designed around punch cards may be incorporated at a later date.

Programme of Work. It is not proposed to limit the programme of work on the machine to any particular type of problem, but to cover a range of problems chosen for their diversity. Thus, in the immediate future, the following items are among those which will be investigated—

1. Partial differential equations arising out of biological calculations.
2. Simultaneous linear differential equations and matrix algebra, and their application to the cotton and aircraft industries, and electricity distribution.
3. Tabulation of Laguerre functions.
4. Design of optical systems.
5. Fourier synthesis for X-ray crystallography.
6. Design of plate fractionating towers.
7. Chess problems.

The authors wish to express their gratitude to the Royal Society, the Ministry of Supply, the Department of Scientific and Industrial

Research, Ferranti Limited, and the General Electric Company Limited for their generous assistance to this project.

REFERENCES

1. Williams, F. C. and Kilburn, T. "A storage system for use with binary digital computing machines." *Proc.I.E.E.* **96,** Pt. II, 30 (1949).
2. Williams, F. C., Kilburn, T. and Tootill, G. C. "Universal high-speed digital computers: a small-scale experimental machine." *Proc.I.E.E.* **98,** Pt. II, 61 (1951).
3. Williams, F. C. and West, J. C. "The position synchronization of a rotating drum." *Proc.I.E.E.* **98,** Pt. II, 61 (1951).
4. Williams, F. C., Kilburn, T. and Thomas, G. E. "Universal high-speed digital computers: a magnetic storage system." *Proc.I.E.E.* (to be published).
5. Kilburn, T. "The University of Manchester high-speed digital computing machine." *Nature* **164,** 684 (1949).
6. Williams, F. C., Kilburn, T. and Robinson, A. A. "Universal high-speed digital computers: serial computing circuits." *Proc.I.E.E.* (to be published).
7. Kilburn, T. "The new universal computing machine at the University of Manchester." *Nature* **168,** 95 (1951).

Chapter 7

CALCULATING MACHINE DEVELOPMENT AT CAMBRIDGE

THE FIRST PURELY ELECTRONIC CALCULATING MACHINE to be put into operation was the E.N.I.A.C., which was built at the Moore School of Electrical Engineering in the University of Pennsylvania and completed in the summer of 1946. Although this machine was very successful, and its construction marked a big step forward, theoretical analysis of the circuit requirements showed that electronic calculating machines of much greater power could be built with a far smaller quantity of equipment. The new ideas were ably presented in a report drafted by Dr. J. von Neumann on behalf of a group associated with the Moore School. By the end of 1946 work was already starting in a number of places on the construction of machines of the new kind. Although the machines would be much smaller than the E.N.I.A.C.—they would have three or four thousand valves instead of 18,000—they would still be quite large and complicated pieces of equipment.

As soon as the possibilities of electronic computing machines came to be appreciated, it was clear that such a machine could play an important part in many fields of scientific research in which work was being done at Cambridge. Much development work would, however, have to be done first. Apart from the technical problems involved in the construction of a machine, there was much that would have to be learnt about the best methods of handling numerical work on the scale that the machine would make possible. Moreover, the theoretical analysis referred to had shown that the most efficient machine from the point of view of economy of equipment would be one that worked entirely in the binary system, the necessary conversion to and from the decimal system being done by the machine itself during the processes of input and output. There were good reasons for believing that such a machine would not be in any material way less convenient to use than a decimal machine, but until practical experience had been obtained no one could say definitely that this was the case. Again, the task of preparing programmes for the machine appeared at first sight to be very tedious and lengthy. If this were so it would mean that the machine would be useful for applications in which one basic programme could be used to obtain a large number of different results—for example in the calculation of mathematical tables—but less useful as a general-purpose computing instrument. It had been suggested that the problem of programming might be simplified and reduced to reasonable proportions if a "library" of short programmes for performing standard operations (what are usually called subroutines) were formed. Clearly if this view were justified much of the value of high-speed computing machines would depend on the existence of such a library, on its completeness, and on the ease with which programmes could be formed from it. A large field of experimentation and development would therefore be opened up as soon as a machine became available.

It was with these ideas in mind that the Electronic Delay Storage Automatic Calculator (E.D.S.A.C.) was planned. Work began in the University Mathematical Laboratory in the early part of 1947 and the machine performed its first fully automatic calculation in May, 1949. It was the first electronic calculating machine of the new kind, complete with high-speed store and input and output devices, to be put into operation.

A few months' experience with the E.D.S.A.C. was sufficient to demonstrate that it was in every way a practical proposition, both from the point of view of the engineer and from that of the mathematical user. During the first nine months of operation the nucleus of a library of subroutines was constructed, and convenient methods of incorporating them in programmes were developed. It was found that once these methods were established the art of programming could be learnt in a comparatively short time, and many research students from other departments in the University who have worked in the laboratory have been able to do their own programming.

The library at present contains about 150 subroutines and is still growing. The first subroutines to be constructed were for such operations as taking square roots and evaluating sines and cosines. (It may come as a surprise to some people to learn that it is usually better, when using an electronic high-speed calculating machine, to calculate sines and cosines afresh whenever they are required rather than to store a table in the machine.) Later it became clear that subroutines could be constructed for performing quite complicated mathematical processes, such as numerical integration or the solution of differential equations. These subroutines provide, as it were, the framework of the calculation, and the programmer constructs auxiliary subroutines which specify the details according to the requirements of his particular problem. It has also been found possible to construct subroutines which help the programmer to assemble a collection of subroutines and a master routine so as to form a complete programme. In fact, once it has been realized that any particular task repeatedly encountered in the work of constructing programmes can be performed by applying fixed rules, there is no reason why a subroutine should not be constructed to perform it. In this way it is to be hoped that as time goes on the labour of constructing programmes will be progressively reduced.

In addition to subroutines the library contains a number of complete programmes for dealing with problems which commonly occur in numerical analysis. These include programmes for solving sets of simultaneous equations, for solving polynomial equations, and also for doing something rather similar to the process known as relaxation.

The E.D.S.A.C. uses ultrasonic delay units for storage, this system being the only one based on principles sufficiently well established in 1947 for it to be adopted with confidence. The store was designed to have 1,024 storage locations, each capable of holding a five-digit decimal number expressed in binary form.

Orders are represented in a numerical code, each order being equivalent to a five-digit decimal number, and are held in the same store as the numbers. It is possible to combine two adjacent storage locations to hold a single ten-digit decimal number.

An ultrasonic delay unit consists of a steel tube 5 ft long, closed at each end by a quartz crystal, and filled with mercury. The pulses which represent the numbers to be stored are applied to one of the quartz crystals and converted into sound waves of very high frequency. These travel through the mercury, taking approximately one millisecond to reach the second crystal which reconverts them into electric pulses. These pulses are amplified and shaped, and sent back to the input crystal of the delay unit. They travel through the mercury a second time, and on emerging are once more amplified, shaped, and sent back to the input. They continue to circulate in this way, passing through the delay unit a thousand times each second, until they are required for use in the calculation. The pulses are about a microsecond long and are separated by a minimum interval of one microsecond. It is thus possible for rather more than 500 pulses to be travelling down the column of mercury at any one

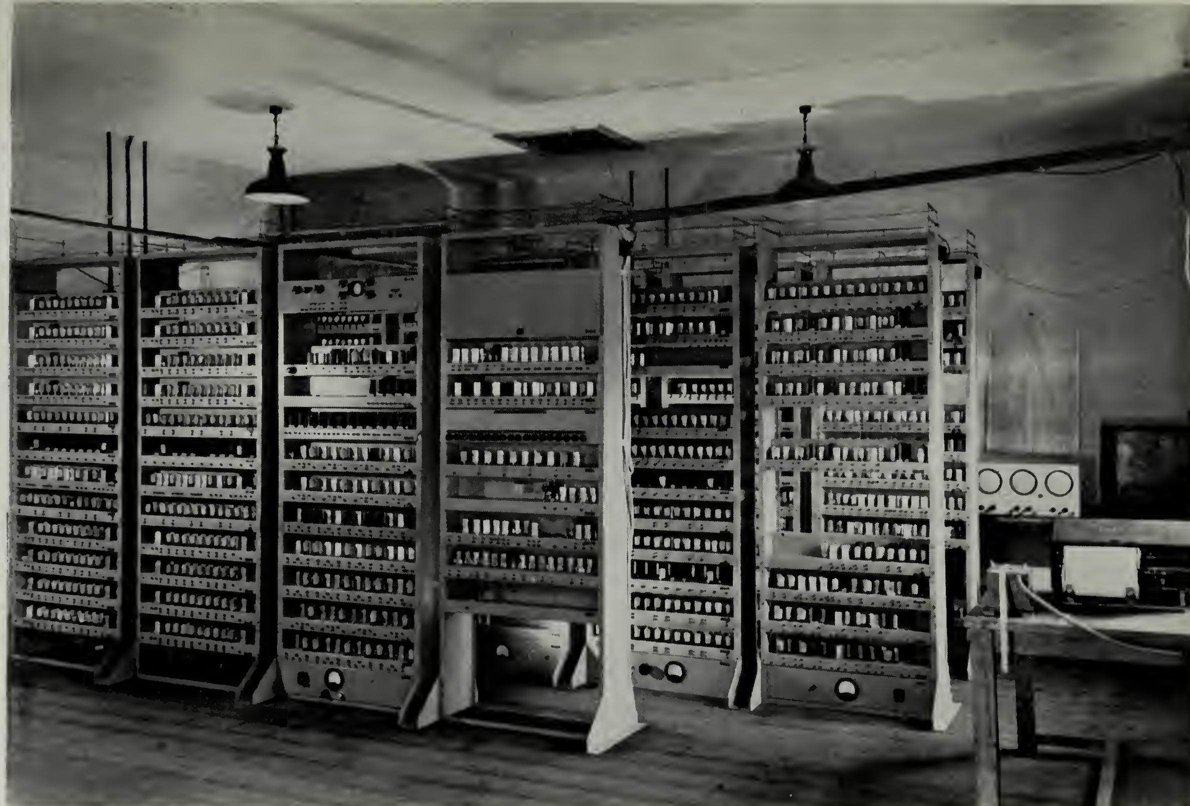


Plate XI. A General View of the E.D.S.A.C.

time. Since each pulse corresponds to a binary digit this means that 32 numbers, of 17 binary digits each (equivalent to about 5 decimals), can be stored in a single tube of mercury, or tank as it is usually called. The tanks are built in two batteries. Each battery consists of 16 tanks and contains 220 lb of mercury.

An idea of the general appearance of the machine may be obtained from Plate XI. It contains about 3,000 valves and consumes 15 kW of power. The order code is of the variety known as *single-address*; that is, each order has reference to at the most one location in the store. For example an order might cause a number standing in a certain location in the store to be added to the number in the accumulator. Simple operations of this kind take on the average 1·5 msec, which includes the time taken for the order and the number to be extracted from the store. Multiplication takes about 6 msec. The E.D.S.A.C. has no automatic divider, and division is performed by means of an iterative procedure; it takes between 120 and 450 msec, depending on the magnitude of the divisor.

Punched paper tape, of the kind used in teleprinter operation, is used for input. The tape is read by means of a photo-electric tape-reader. The orders required for performing a calculation, together with any numbers which may also be required, are punched on a single tape, or, as is sometimes more convenient, on separate tapes which are placed one after the other in the tape-reader. Conversion of the addresses of the orders and the numbers to binary form is done automatically as part of the programme, and it is not necessary for the programmer to do any conversion of this kind. No setting-up of the machine, apart from putting a new tape in the tape-reader, is necessary when changing over from one problem to another. The machine may therefore be used to obtain, during the course of a single day, results relating to a large variety of different problems. The results are printed on a Creed teleprinter.

The diversity of the applications of the machine which have been made so far is illustrated by the following list, which is not intended to be complete—

Theoretical Chemistry. The calculation of atomic and molecular wave functions.

Crystallography. Two and three dimensional Fourier synthesis.

Radiophysics. The equations of propagation of radio waves in the ionosphere.

Geophysics. The motion of a pendulum mounted on an unsteady platform; this has application to the measurement of gravity at sea.

Astrophysics. Theoretical studies of the stability of a star.

Statistics. Serial correlation coefficients.

Mathematical Tables. Complex gamma functions, complex error integrals and prime numbers.

Further information about the construction of the E.D.S.A.C. will be found in references 1 and 2. A detailed account of the library of subroutines and of the methods used for preparing programmes will be found in reference 3.

The initial success of the E.D.S.A.C. has served to stimulate fresh development in the laboratory. The performance of the E.D.S.A.C. itself has been continually improved, and its scope enlarged, by many modifications to the original design. The most important of these has been the provision of an auxiliary store with capacity for many thousands of numbers based on the use of magnetic tape. The construction of an entirely new machine with many original features is in progress. At the same time, experience in the solution of problems is accumulating, both at Cambridge and elsewhere, and the outlines of a new branch of mathematical study which might be called "automatic computing" are beginning to emerge.

REFERENCES

1. Wilkes, M. V. and Renwick, W. "The E.D.S.A.C.—an electronic calculating machine." *J. Sci. Instr.* **26,** (1949) 385.
2. Wilkes, M. V. and Renwick, W. "The E.D.S.A.C. (electronic delay storage automatic calculator)." *M.T.A.C.* **4,** (1950) 61.
3. Wilkes, M. V., Wheeler, D. J. and Gill, S. *The preparation of programs for an electronic digital computer, with special reference to the E.D.S.A.C. and the use of a library of subroutines.* Addison-Wesley Press Inc. (Cambridge, Mass. 1951).

Chapter 8

AUTOMATIC COMPUTATION AT THE NATIONAL PHYSICAL LABORATORY

One of the most modern digital computers which embodies developments and improvements in the technique of automatic electronic computing was recently demonstrated at the National Physical Laboratory, Teddington, where it has been designed and built by a small team of mathematicians and electronics research engineers on the staff of the Laboratory, assisted by a number of production engineers from the English Electric Company, Limited. The equipment so far erected at the Laboratory is only the pilot model of a much larger installation which will be known as the Automatic Computing Engine, but although comparatively small in bulk and containing only about 800 thermionic valves, as can be judged from Plates XII, XIII and XIV, it is an extremely rapid and versatile calculating machine.

The basic concepts and abstract principles of computation by a machine were formulated by Dr. A. M. Turing, F.R.S., in a paper(1) read before the London Mathematical Society in 1936, but work on such machines in Britain was delayed by the war. In 1945, however, an examination of the problems was made at the National Physical Laboratory by Mr. J. R. Womersley, then superintendent of the Mathematics Division of the Laboratory. He was joined by Dr. Turing and a small staff of specialists, and, by 1947, the preliminary planning was sufficiently advanced to warrant the establishment of the special group already mentioned. In April, 1948, the latter became the Electronics Section of the Laboratory, under the charge of Mr. F. M. Colebrook.

The basic principles of all automatic computing machines, including the A.C.E., which are capable of performing extended calculations unaided, have already been explained. In Fig. 8/1, the various sections of the equipment and their functions are represented schematically. The central organ of the machine is the control unit which takes the place of the operator in the desk calculation. Instructions to the machine are fed into the input, which may, as in the case of the A.C.E., be standard Hollerith equipment. The instructions are in code form; for example, in the case of Hollerith equipment, they consist of holes punched in a variety of patterns on a pack of cards, like the ones shown in Plate III. Some of the instructions represent numbers on which various arithmetical operations have to be performed; others define the operations themselves, including the operation of transfer to or from one or other of the units of the machine. Still other instructions may represent tables of numerical values, which must be stored up till required, or information to the

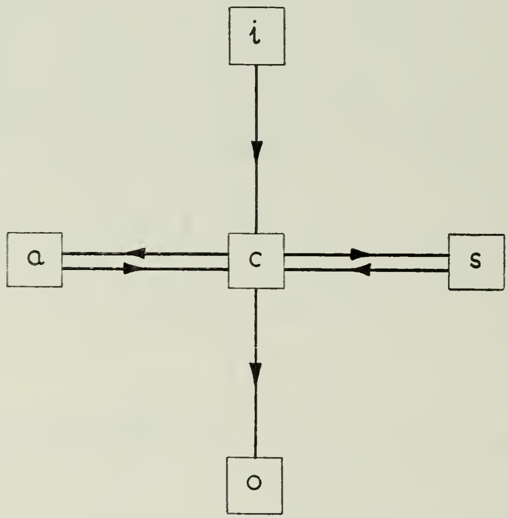


FIG. 8/1. A schematic diagram of the A.C.E.

machine which will enable it to work out such values when they are needed. Finally, the machine must be given instructions which will enable it to exercise judgment in the manner which has already been described and to pass the result to the output when the calculation is finished.

It is clear, therefore, that in addition to possessing an arithmetical unit equivalent to the desk calculating machine, in which the actual calculation is performed, the automatic computing machine must contain a store or memory unit to perform the functions of the paper, pencil, and book of tables, and to house instructions for the control unit. This, indeed, is one of its most important parts. In the course of any calculation on such a machine, moreover, information is continually flowing to and fro between the store and the arithmetic and control units.

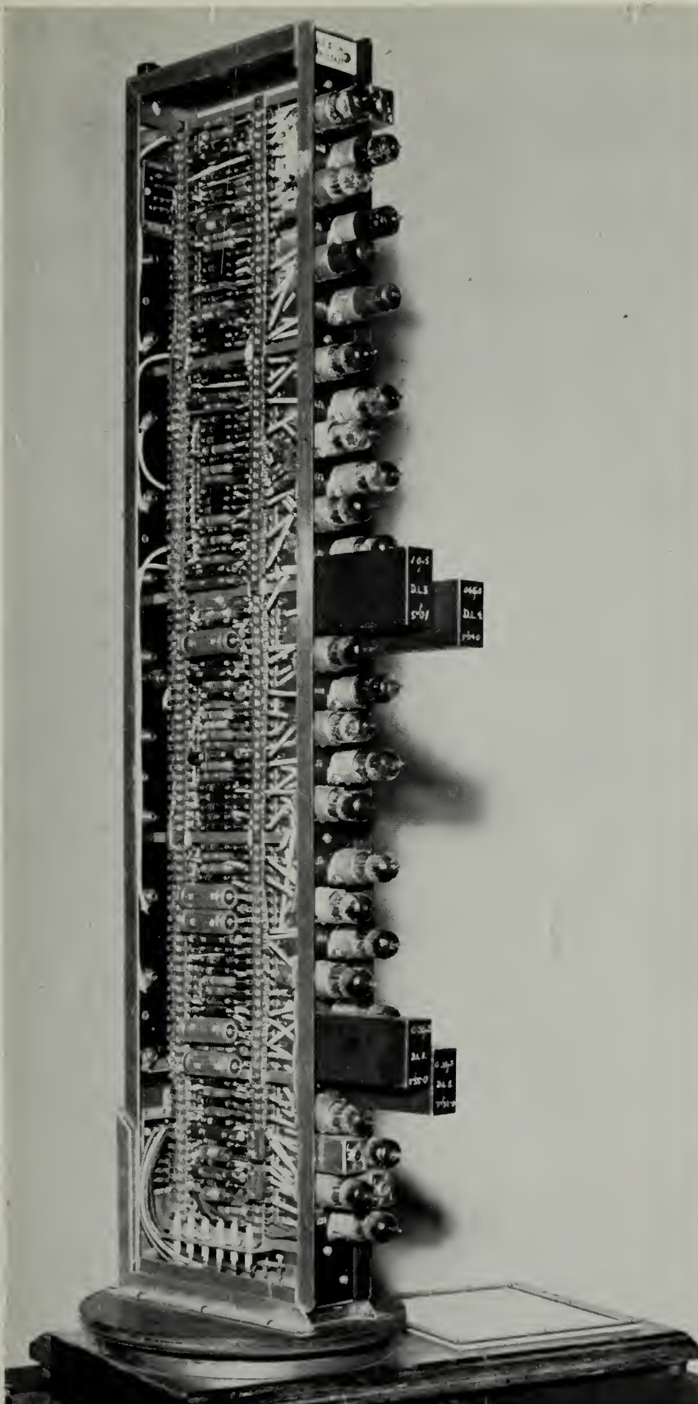


Plate XII. One Unit of the A.C.E.

Each stage of the calculation is accomplished at prodigious speed so that storage space and the information and instructions in storage must be available practically all the time. To make them available continuously, however, is impracticable, as this would entail the use of an excessively large amount of equipment. In consequence, the A.C.E. employs dynamic storage, in which information and instructions are circulated continuously round delay lines until no longer required.*

In the A.C.E., the master oscillator produces pulses at micro-second intervals and it is possible, therefore, to produce a train of pressure waves in the mercury inside the "tanks" with a spacing of approximately 0·06 in. between successive fronts. Two sizes of delay unit are used in the present equipment. In the smaller, the mercury column is approximately two inches long and three of them may be seen in Plate XIII; the operator is adjusting one of them. Each of these units can accommodate 32 pulses, and the longer units, which cannot be seen in the illustration, can accommodate 1,024 pulses. Decimal numbers of 9 digits can be represented in the binary scale by numbers of not more than 32 digits, so that each of the short delay lines can accommodate, in binary form, any number of up to 9 decimal digits together with an indication of its sign, while each of the long delay lines can accommodate 32 such numbers. There are also lines 65 digits in length, used to contain the product of 2 thirty-digit numbers.

By means of the delay system, therefore, any number of not more than 9 decimal digits can be circulated continuously for as long as is necessary and each of its binary digits becomes available, if required, every 32 μsec which is the time required for each stage of a multi-plication. In the course of a calculation, a train of pulses may be required to circulate in the store many thousands or even millions of times. If the same pulse train were merely shunted continuously round the delay line it would soon become unrecognizable owing to degeneration of the waveform. It is for this reason that the electronic gate is employed as already described, to enable a pulse train to create itself again after each circulation. It might be thought that reflections of the waves within the tube would interfere with the wave pattern, but, in fact, the attenuation of the waves in the mer-cury, which, in the short tubes, is increased by the insertion of a wire gauze in the tube, is sufficient to obviate trouble of this kind.

Details of the circuitry of the A.C.E. model have not yet been

* See pages 59 and 132 for detailed description.

disclosed. As was mentioned earlier, the input and output consist of standard Hollerith punch-card equipment, some of which is visible in Plate XIV. Although comparatively rapid by ordinary standards, this equipment operates relatively slowly compared with electronic apparatus. At the time of construction of the automatic computing engine, however, the Hollerith equipment was the fastest commercially-produced equipment of the kind available. The final design may employ other means, but it was considered undesirable to divert effort and attention from the main problem in the early stages. Another way in which the A.C.E. will be improved in future will be by the provision of greatly increased storage capacity; it is intended to equip it with a magnetic storage system.

Reference has already been made to the speed of the A.C.E. The present model can deal with nine hundred million binary digits in a quarter of an hour and is able to multiply any two numbers of nine decimal digits each in less than one five-hundredth of a second. It is also extremely versatile. Although it calculates with numbers in the binary scale, it is unnecessary for it to be provided with numerical data in binary form. Numbers may be supplied to the equipment in decimal form, whereupon, when instructed to do so, it will convert them into binary form, complete the calculation and turn the answer into decimals, which the Hollerith equipment will translate automatically into punchings on a set of cards. Auxiliary Hollerith equipment may then be used to turn the punchings into printed numerals. The work of programming and coding normally takes much longer than the calculation of a result, but once completed, the requisite punched cards may be used repeatedly for the same type of calculation. When the A.C.E. equipment was demonstrated, it was engaged in computing the paths of rays through a lens system consisting of several components. This is a calculation which arises in the design of optical systems and, when several refracting surfaces are involved, it presents a complicated problem since the optimum shape and position of each component must be determined. In the past, it has been necessary to make an informed guess at what is likely to be a satisfactory arrangement, and to test it by working out the paths of a few rays, but the process is extremely tedious and the best solution is not always found in the end. The A.C.E. is able to calculate the paths of a large number of rays in a few minutes. Approximately a week was required in this case to complete the programming and coding of the instructions, but the latter are now available for any similar problems in the design of lens systems.

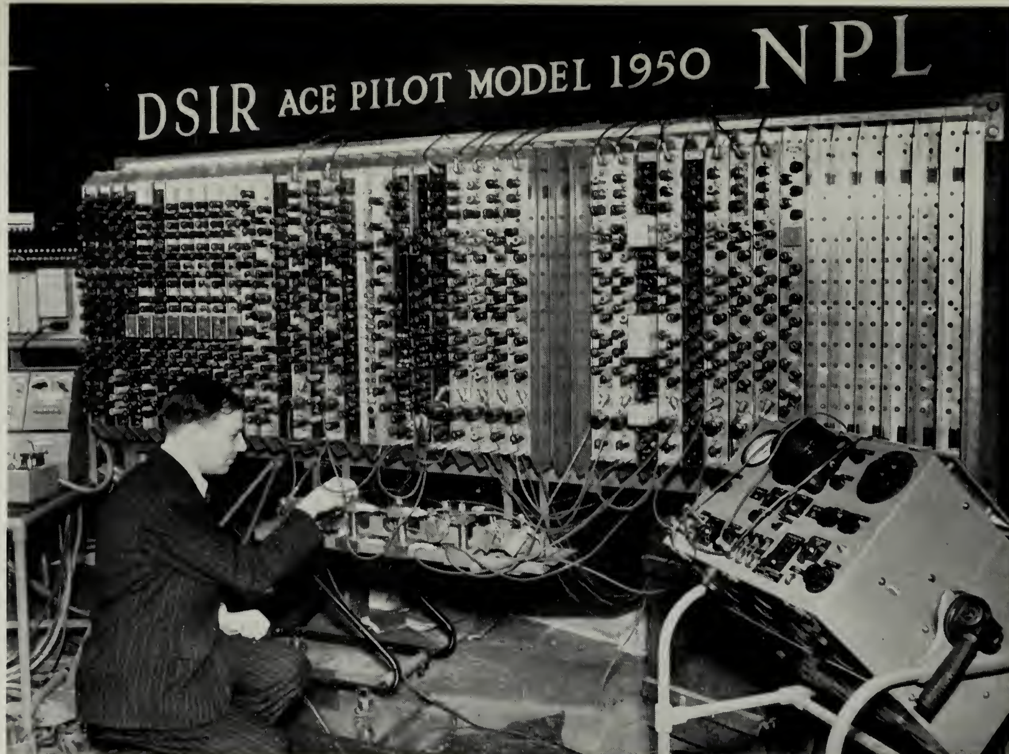


Plate XIII. A View of the A.C.E. showing Delay Units

Plate XIV. A View of the A.C.E. showing the Hollerith Equipment used for Input and Output

To err is human, and all who have had experience of lengthy numerical calculations know that the work demands constant alertness and attention to details if mistakes are to be avoided. Automatic computing machines too, though not subject to the distractions which beset the human operator, are inclined, at times, to be wayward and to make mistakes owing to failure of some part of the equipment. Fortunately, when failure occurs, it is generally sufficiently serious for the effects to be obvious at once, and on most other occasions it is fairly easy to check the accuracy of performance by means of a test calculation which has been worked out independently. The problem of failure is, however, an important one and must be considered in designing the equipment. Electronic valves and components are surprisingly reliable, but failures do occur occasionally, and when such items are employed in hundreds or even thousands and tens of thousands, their overall reliability becomes subject to the laws of probability and can be assessed on a statistical basis. One can envisage a stage, therefore, in the growth of an automatic calculating machine, where a high probability would exist of the equipment being permanently out of order; and at some earlier stage breakdowns might occur with sufficient frequency to make the equipment of little value (but see also Chapter 4). To attempt to diagnose faults by testing individual components *seriatim* would be a futile procedure on equipment like the A.C.E. All valves and components are, of course, tested initially before assembly, but the equipment is assembled as a series of units, one of which is illustrated in Plate XII. There are forty such units in the pilot model of the A.C.E. but only four of them do the arithmetic, the remainder performing the functions of control and storage. Should failure of the equipment occur, the faulty unit can be quickly isolated and replaced by a spare, after which the unit which has been removed can be tested and repaired. In this way, the equipment can be kept in good order.

REFERENCE

1. Turing, A. M. *Proc. London Mathematical Society*, (Ser. 2) **42**, (1936) 230. Correction appears in *Proc. London Mathematical Society*, (Ser. 2) **43**, (1937) 544.

Chapter 9

THE HARWELL ELECTRONIC DIGITAL COMPUTER

THE HARWELL ELECTRONIC DIGITAL COMPUTER[1] is not intended to undertake enormous computations too lengthy to carry out by other means. It is intended rather to do the work of a few operators with desk machines where the work to be done is of a routine or repetitive nature. As a result, simplicity and reliability have been regarded as of more importance than speed of operation, and the machine performs the operations of addition, subtraction, multiplication and division little faster than a desk machine. Apart from speed, however, the computer provides most of the facilities found in the larger, faster machines.

Relays are well suited to the switching of a multiplicity of circuits and they are therefore used for control and for routing information. To minimize the storage capacity needed, all orders and other data are normally read off a perforated paper tape as they are needed in the calculation. Numerical and other data obtained in the course of the calculation must, however, be stored in the machine. Use of relays for this purpose is not economical and Dekatrons[2] have therefore been used. These are cold-cathode gas discharge tubes having ten cathodes and a common anode. The discharge flows from the anode to any one cathode and may be stepped from one cathode to the next by application of suitable pulses. Each tube can therefore store one decimal digit. Each store uses 9 Dekatrons, providing 8 decimal digits and a sign. There are facilities for using up to 90 such stores, though, at the time of writing, only 40 are being used. The larger number is expected to be adequate for most purposes, since very few orders will normally be stored in the Dekatrons.

The Dekatrons are compact and use negligible power. Their use also simplifies routing, since parallel operation is relatively simple, and numbers in two stores may be added or subtracted merely by transferring the contents of one store directly into the other in a single operation.

The computer contains about 380 relays, 18 Dekatrons, 80 thermionic valves and 40 cold-cathode triodes, plus 18 relays and

90 Dekatrons per group of ten stores. The total power consumption is less than one kilowatt. It has been in use by the Computing Group at Harwell since May, 1952, and is frequently run unattended overnight and over week-ends.

Control and Routing

The computer works on a two-address principle; each arithmetic order consists of 5 decimal digits which specify the operation to be performed and the sending and receiving addresses. There are also orders for changing the source from which subsequent orders are to be drawn, selecting one of the ten alternative layouts for the printed results, and calling for attention. Orders can, if desired, be stored in any of the Dekatron stores, subjected to arithmetic operations if necessary, and drawn from successive addresses for use. At any point in the programme the contents of any storage location may be fed out to one or more page printers (modified teleprinters) or tape perforators. The tape from a perforator may be fed back to a tape reader at the input to the computer, thus providing a restricted form of long-term storage.

Arithmetic Operations

Each eight-digit number and its sign are stored in 9 Dekatrons with the decimal point fixed and the numbers stored lying between + 10 and − 10. Between each of the 10 cathodes of the Dekatron are two guide electrodes, and by making these two electrodes in turn negative with respect to the cathodes the discharge is transferred from one cathode to the next. This feature of requiring pulses in the correct time relationship on two guide electrodes has been used to reduce the number of routing relay contacts associated with each store. A digit is stored in a Dekatron by applying to the guide electrodes the correct number of pulses to step the discharge to the required cathode. Only the zero cathode is separately connected and the method of extracting a stored digit therefore depends on a signal generated when the discharge reaches this cathode.

The basic arithmetical operation in the computer is the transfer simultaneously of digits from one storage location to another. Every tube of the sending store can be fed with a train of ten pulses so that the discharge circulates and returns to its original position. The output cathode of each tube of the sending store is connected through a relay "shifting" network to a transfer circuit. The latter also receives a train of 10 pulses which it routes to one output until the sending tube passes its zero cathode, and then switches the remainder of the train to a second output. The number of pulses appearing at the second output is equal to the number in the sending tube, while at the first output there appears the complement on ten of this number. The complement train is fed to the receiving tube for subtraction, and the number train for addition. If the information

TABLE VI

EXTRACT FROM THE SOLUTION OF A DIFFERENTIAL EQUATION

$$\frac{1}{k^2} y \cdot y'' = -y^2\beta^2(x) + \frac{1}{1 - \mu x}$$

where $\beta(x)$ is a given numerical function

μ − 0·12500 $x/10$	k^2 + 0·06710 y	y' at $x = 0$ − 0·30000 y''	Difference Check
+ 1·10000	+ 0·45889	+ 0·03470	+ 0·00000 00
+ 1·11000	+ 0·46584	+ 0·03291	+ 0·00000 00
+ 1·12000	+ 0·47312	+ 0·03114	+ 0·00000 00
+ 1·13000	+ 0·48071	+ 0·02936	+ 0·00000 04
+ 1·14000	+ 0·48859	+ 0·02756	+ 0·00000 01
+ 1·15000	+ 0·49675	+ 0·02574	+ 0·00000 01
+ 1·16000	+ 0·50517	+ 0·02393	+ 0·00000 01
+ 1·17000	+ 0·51383	+ 0·02211	+ 0·00000 02
+ 1·18000	+ 0·52271	+ 0·02029	+ 0·00000 02
+ 1·19000	+ 0·53179	+ 0·01849	+ 0·00000 02
+ 1·20000	+ 0·54106	+ 0·01668	+ 0·00000 01
+ 1·21000	+ 0·55049	+ 0·01489	+ 0·00000 01
+ 1·22000	+ 0·56007	+ 0·01311	+ 0·00000 01
+ 1·23000	+ 0·56979	+ 0·01135	+ 0·00000 02
+ 1·24000	+ 0·57961	+ 0·00960	+ 0·00000 02
+ 1·25000	+ 0·58954	+ 0·00788	+ 0·00000 02
+ 1·26000	+ 0·59954	+ 0·00617	+ 0·00000 02
+ 1·27000	+ 0·60960	+ 0·00448	+ 0·00000 01
+ 1·28000	+ 0·61971	+ 0·00281	+ 0·00000 02
+ 1·29000	+ 0·62984	+ 0·00117	+ 0·00000 01
+ 1·30000	+ 0·63999	− 0·00044	+ 0·00000 02
+ 1·31000	+ 0·65014	− 0·00204	+ 0·00000 01
+ 1·32000	+ 0·66026	− 0·00361	+ 0·00000 01
+ 1·33000	+ 0·67035	− 0·00515	− 0·00000 00
+ 1·34000	+ 0·68038	− 0·00666	− 0·00000 00
+ 1·35000	+ 0·69035	− 0·00814	+ 0·00000 01
+ 1·36000	+ 0·70024	− 0·00959	+ 0·00000 01
+ 1·37000	+ 0·71003	− 0·01101	+ 0·00000 01
+ 1·38000	+ 0·71971	− 0·01241	− 0·00000 00
+ 1·39000	+ 0·72927	− 0·01377	+ 0·00000 01

See pages 8 and 346.

in the sending store is not required for subsequent use the train of pulses used for circulation may be derived from the complement outputs of the transfer circuits so that each tube only circulates as far as its zero cathode.

A carry circuit is associated with each transfer circuit. If the corresponding receiving tube passes zero during a transfer, the carry circuit remembers that a carry-over is needed, and routes a pulse to the next more significant digit of the receiving store after the transfer has finished. This may initiate a further carry-over and carry pulses are generated until no more are required.

Addition and subtraction involve two storage positions and do not require the use of the accumulator. Multiplication by each digit of the multiplier in turn requires repetitive addition (or subtraction) of the multiplicand. Storage locations are needed for the multiplier and multiplicand and for the accumulation of the product. Since the product contains more digits than either multiplier or multiplicand, a long store of 15 digits is provided for use as an accumulator. Division, by the usual long-division process, uses this accumulator to contain the dividend, and normal storage locations for the divisor and for the formation of the quotient. The selection of each digit of the multiplier, and the routing of the multiplicand into the accumulator through the shift network, is controlled by sequence relays, but the repetitive transfers are controlled electronically. Multiplication of 2 eight-digit numbers takes between five and ten seconds.

Table VI illustrates the presentation of results which were obtained in a typical computation. Each line involved about five minutes' work by the computer.

REFERENCES

1. Barnes, R. C. M., Cooke-Yarborough, E. H. and Thomas, D. G. A. *Electronic Engineering*. (Aug., Sept., 1951).

2. Bacon, R. C. and Pollard, J. R. *Electronic Engineering* (May, 1950) 173.

Chapter 10

THE TELECOMMUNICATIONS RESEARCH ESTABLISHMENT PARALLEL ELECTRONIC DIGITAL COMPUTER

This computer, being designed and built at the Telecommunications Research Establishment, is parallel operated for twenty-four binary digits with a fixed binary point; it uses a one address code, and cathode-ray-tube storage based on the principle invented by Professor F. C. Williams (see page 60) but specially adapted for parallel operation. As far as possible both noughts and ones are represented by active states.

A magnetic drum, with a maximum storage density of sixteen thousand digits per square inch, is to be used as a large capacity *outer store.* All the mathematical operations are carried out in a single "Relation Unit."

A one-digit pilot machine has been designed and built, and has successfully carried out routines of operations continuously over periods of hours. Each operation is performed in 20 μsec as will be the case in the complete twenty-four digit machine.

Design Principles

As far as possible the binary digits 0 and 1 are both represented by active states or occurrences. The absence of a voltage, or charge, is not taken to represent 0. By this means the machine itself can distinguish between the "bit" of information 0, and the failure of a 1 to be received, and will be able to indicate where a fault has occurred.

As a corollary all circuits are designed as far as possible so that component failures never cause either of the two meaningful states 0 and 1. As an example, the coincidence circuit of Fig. 10/1 (*a*) causes the anode to rise if the point *A* rises and the point *B* falls, but this meaningful state of a cut-off valve also occurs if the heater fails. On the other hand, in the cathode-follower circuit of Fig. 10/1 (*b*) the cathode rises if *A* and *B* rise; this meaningful state does not occur if the heater fails. Using the terminology of the railway signal engineer, circuit 10/1 (*a*) can give "wrong-side failures," circuit 10/1 (*b*) can give only "right-side failures."

In registers, for each digit a 0 and a 1 trigger circuit are used, each for a meaningful and a failure state. A built-in check circuit gives no indication if one and only one of these two triggers is operated.

The present storage system does not possess this property, but it is hoped at a later stage to work on this problem.

General Description. The machine is to be parallel-operated using cathode-ray-tube storage. The accuracy will be to 23 binary places or about 6 decimal places. The essential components are shown in Fig. 10/2.

The outer store will use a magnetic drum to store instructions

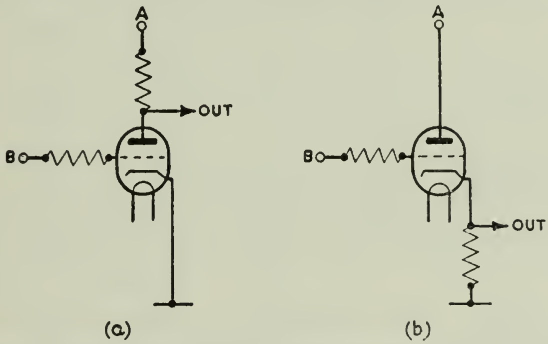


FIG. 10/1. "Wrong-side failure" and "Right-side failure" circuits

and results in binary form using 24 write/read heads operating in parallel. Other heads will provide basic waveforms which time the operation of the whole computer. Conversion to binary form will be effected by the computer itself.

The inner store consists of 24 cathode ray tubes each storing one binary place for 512 numbers. The twenty-fourth digit place (most significant) is the sign digit.

The accumulator register stores partial sums. Addition is carried out by sending the contents of the accumulator and the inner store simultaneously to the relation unit, *R.U.* Instructions are also sent to *R.U.* from the inner store and can be modified by the contents of *A*. The output of the relation unit is stored in the shift register. To complete the operation the contents of the shift register are transferred to the accumulator. It is also possible by use of the shift register to shift the contents of the accumulator by one binary place in the direction of less significance (i.e. to divide by 2).

The position being used at any instant on the cathode ray tube is determined by one of three registers—regeneration, instruction, and

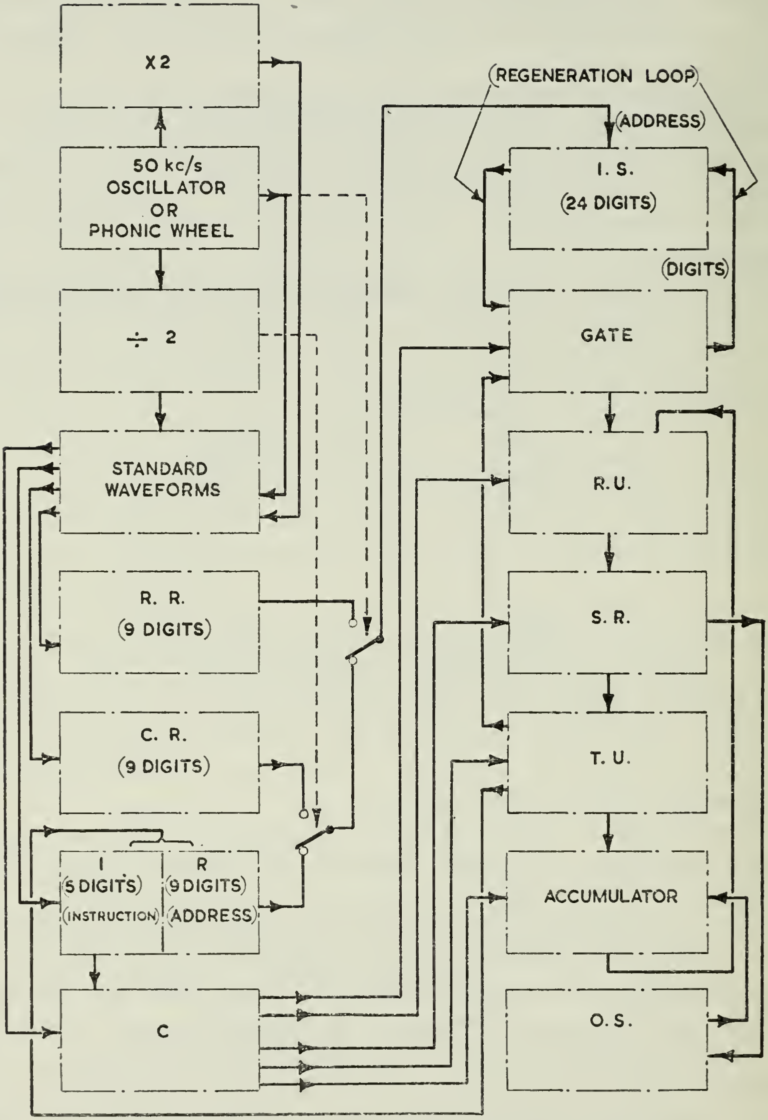


FIG. 10/2. Block schematic diagram of complete computer control—the contents of the register being used at any instant being converted into analogue form as a deflection voltage.

R.R. is the regeneration register which is a scale-of-two counter of 9 stages counting regularly and so causing all digits of the cathode ray tube to be regenerated sequentially regardless of how or when they are used mathematically.

Alternating between moments of regeneration are moments of action when instructions are either being read or being carried out. The deflection plates are then connected to *C.R.* and *I.R.* respectively.

C.R., the control register, contains the address of the current instruction, i.e. the number of the instruction in the list of instructions. There are 9 places in this register.

I.R., the instruction register, contains the current instruction which consists of the address (in the inner store) of the number being used (9 binary digits) and the type number of the instruction to be carried out (5 digits).

The Control accepts the instruction type in coded form from the 5 most significant digits in *I.R.*, and causes the different transfer gates to operate in their correct order, the whole series being related to a basic "clock" waveform.

Types of Instruction. The twenty-four digit machine will not include initially either a magnetic store or a multiplication organ; until these are added the list of instructions will be as follows—

1. (n) to A. Clear accumulator and add to it the content of address n of inner store (i.e. write the contents of n into accumulator).

2. $-(n)$ to A. Clear accumulator and subtract from it the content of address n of inner store.

3. $(A) + (n)$. Add content of address n to content of accumulator and place answer in accumulator.

4. $(A) - (n)$. Subtract content of address n from content of accumulator and place answer in accumulator.

5. (A) to n. Transfer content of accumulator to address n of store. The content of the accumulator is unaltered.

6. R. Shift content of accumulator one place to the right, i.e. divide by two.*

7. (A) & (n). Compare contents of accumulator and address n of the store, digit by digit; wherever there is a one in the accumulator and at n, place a one in the accumulator.

8. (A) &/or (n). Compare contents of accumulator and address

* In the T.R.E. machine, the left-hand end of the accumulator is regarded as the most significant.

n, digit by digit; wherever there is a one in the accumulator, or in n, or in both, place a one in the accumulator (this corresponds to $A \vee (n)$ for the Manchester machine).

9. (A) or (n). Compare contents of accumulator and address n, digit by digit; wherever there is a one in the accumulator or in n, but not in both, place a one in the accumulator (this corresponds to $A \not\equiv (n)$ for the Manchester machine).

10. T to n. If content of accumulator is negative, proceed to instruction stored at address n.

11. J to n. Proceed to instruction stored at address n.

12. I,n. Input. Read the next row of holes on the tape and place them as the five least-significant digits at address n.

13. P (n). Output. Print the character set by the five most-significant digits stored at n.

14. Stop.

The Inner Store

Digital information in the inner store (*I.S.*) is held in the form of electrostatic charges on the fluorescent screen of a commercial type cathode ray tube (C.R.T.), using the defocus-focus method described by Williams and Kilburn,[1] applied in a manner suitable for parallel operation.

The computer will deal with numbers each consisting of 24 binary digits. When the machine demands a number from the store, 24 separate wires have to carry this information simultaneously. It is therefore necessary to have 24 separate C.R.T. stores. The digits stored in any particular store will all have the same degree of significance. The deflection systems of the 24 cathode ray tubes are controlled from a common source, i.e. the digits of a particular number are all stored in similar positions on the screens of their appropriate cathode ray tubes. For the purpose of this description it is necessary to consider only one particular store, since all 24 are identical.

The Problem of Applying C.R.T. Storage to a Parallel Computer. Initial experiments to adapt C.R.T. storage to operate in conjunction with a parallel computer used what Williams and Kilburn[2] have called the dot-dash method. It was soon found that although the dot-dash method was suitable for computers using sequential access it was not satisfactory for a computer working in the parallel mode.

The positive charge of an element in the store is destroyed mainly by secondary electrons from other elements of the same store, particularly adjacent elements. The extent of this interaction is

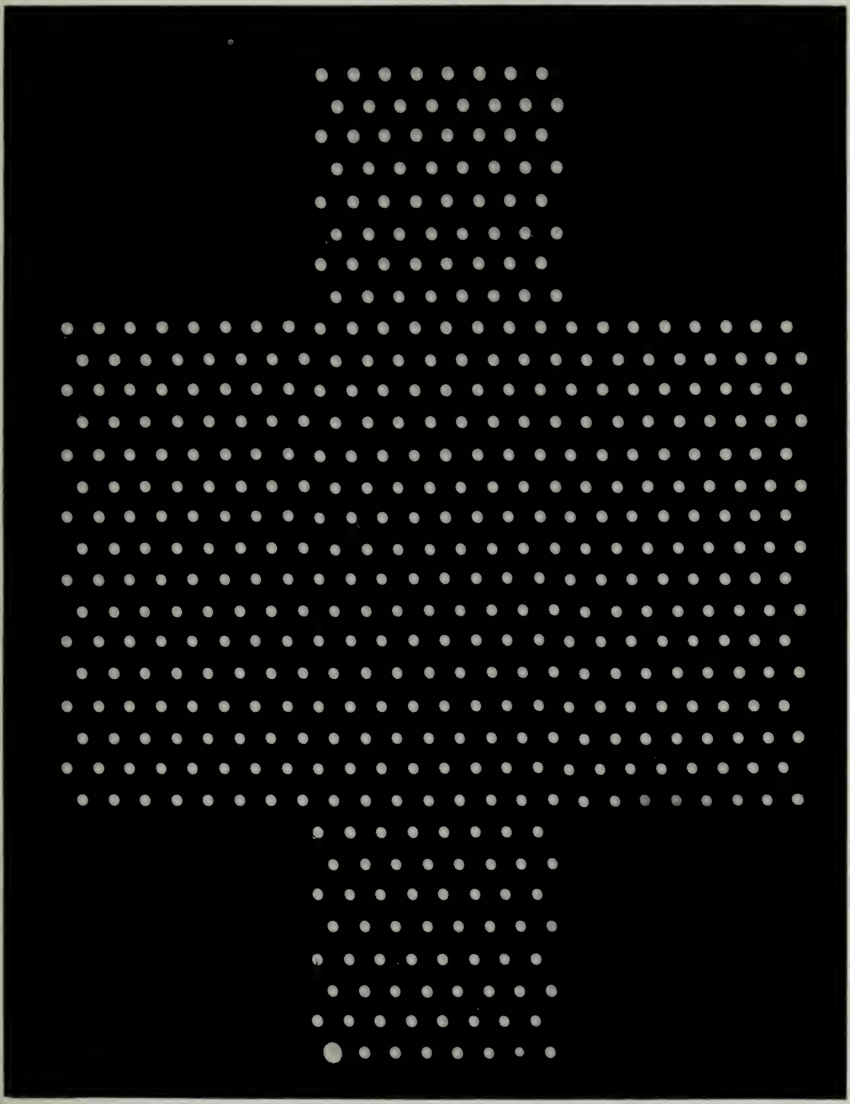


PLATE XV. A CATHODE-RAY-TUBE STORE PATTERN

determined by the net beam current flowing to these elements, by the separation between elements, and by the potential to which the elements of the store are charged.

When operating with a parallel computer, which is given complete freedom in the selection of numbers, the net beam current to adjacent elements can be extremely high compared with the worst case when a computer uses sequential forms of access. Suppose we have a serial machine using n cathode ray tubes; a number of n digits will be stored at n points of a single line on the tube which is scanned serially. Adjacent lines store different numbers. *If* there are also n lines, and one number is required continuously in a computation, then its line will be regenerated n times compared with its neighbour, and this ratio will determine interaction and the spacing which is needed to ensure perfect operation of the store.

In a parallel machine of the same total capacity of n^3 digits in n cathode ray tubes, a single number will be stored at corresponding points on the n tubes. Again, if this number is required continuously then each of its points will be regenerated n^2 times compared with its neighbour. *It will be seen how much more serious the interaction problem is here than in a serial machine.*

From this example, with parallel storage the effect of interaction is seen to be at least n times worse than the equivalent serial case, as a result it is necessary to increase the separation between elements and so reduce storage capacity. In fact using dot-dash methods it was possible to store only about 200 digits on a cathode ray tube.

The disadvantage caused by the extreme sensitivity of the dot-dash system to blemishes of the fluorescent screen means that the C.R.T. final anode voltages have to be limited to 1 kV to prevent these blemishes affecting stored information. Even then only a small fraction of the tubes tested were satisfactory for use as stores. Unfortunately, this limitation of final anode voltage prevents advantage being taken of reduced interaction obtainable by the use of a higher voltage.

The Advantage of the Defocus-focus Method. After various intermediate experiments it was realized that the defocus-focus method had a great many advantages over methods previously used (see page 61); these may be summarized as follows—

(*a*) The refilling process by means of a focused central spot is more efficient than that of a displaced dash, since electrons emitted in all directions contribute to the re-establishment of the charges instead of only those emitted in a particular sector.

(*b*) Since the defocused detection spot is concentric with the focused refilling spot, the chance of a charge barrier, due to a blemish of the screen, preventing re-establishment of the charge is very slight.

(*c*) The state of charge of an element is detected by a defocused spot, and is therefore less sensitive to errors in the deflexion system than when using a focused beam.

(*d*) There should be an improvement in signal strength due to the increased quantity of charge stored in a defocused spot.

Experiments have confirmed these advantages.[3] The duration of the refilling pulse can be reduced from 2·5 μsec to 0·8 μsec, thus reducing the net beam current and the consequent interaction.

Cathode ray tubes that had not been suitable for dot-dash stores with 1 kV final anode voltage, have proved to be perfectly satisfactory with 2 kV final anode voltage, using the defocus-focus method. With 2 kV final anode voltage, one can use that part of the screen secondary-emission characteristic which gives optimum operation conditions. Increasing the final anode voltage of the C.R.T. reduces the potential of the charged elements, with a consequent reduction of signal strength; however, the increased quantity of charge stored in the larger area of the defocused spot compensates for the reduction of potential.

An experimental C.R.T. store of 512 digits using the defocus-focus method has been functioning for about six months, the elements being arranged with 2·5 mm separation. In this pattern 512 digits can be arranged most nearly in the basic circular shape of a C.R.T. screen.

The Address of Numbers in the Store. The rhythm of the store is split into alternate periods of *action* and *regeneration* each of 10 μsec duration. During the action period the computer has control of the cathode-ray-tube deflexion system, and may select any element of the store. During the regeneration period the deflexion system is controlled by a scale-of-two counter, the *regeneration register* (*R.R.*), which allows each element of the store in turn to be selected and regenerated. The first half of each of these 10 μsec periods is allowed for the deflexion system to set up the address of the element, the deflexion voltages then stay steady for the remaining 5 μsec period.

The address selection during the *action period* is determined alternately by one of two registers, the *control register* (*C.R.*) and the *instruction register* (*I.R.*). *C.R.* selects the address of the next instruction to be carried out. This register has the property that, while it is normally a scale-of-two counter, in the case of a conditional instruction, *C.R.* can be made to accept a new configuration determined by the computer. *I.R.* sets up the address associated with the instruction being carried out, this information being obtained via the computer from the contents of the address selected previously by *C.R.*

Control

The instructions contained in the store are in the form of fourteen-digit binary numbers and only differ from numerical data in the manner in which they are used in the machine. The first 9 (least significant) digits define the *I.S.* address of a number which is to be used. The last 5 (most significant) digits define the function to be carried out.

The carrying out of an instruction within the machine may be considered to take place in four distinct beats. The operation performed during each of these beats is as follows—

Beat 1. Set up the control register (*C.R.*) to contain the address of the current instruction (except for a conditional instruction this consists of adding 1 to the control register). Transfer the current instruction to *S.R.*

Beat 2. Transfer the instruction from *S.R.* into the instruction register. The instruction register then initiates the operation of circuits which open the appropriate gates for carrying out the instruction.

Beat 3. Carry out the operation demanded by the current instruction, usually via *R.U.*

Beat 4. Transfer the contents of *S.R.* to the accumulator, or to some other register.

As has already been mentioned, each digit in the various registers is stored by means of a pair of trigger circuits, for 0 and 1 respectively. The transfer of digits from one register to another, and to and from the store is effected on either one wire or the other of a *two wire system* for 0 and 1 respectively.

If each trigger is described as either *off* or *on*, then a digit place is cleared by turning both triggers *off*. For every transfer of a digit to a register the corresponding digit place is first cleared and then the appropriate trigger (0 or 1) is turned *on* under the control of the information source.

It follows that beats 3 and 4 above break up into four moves, for example—

3 (*a*). Clear *S.R.*

3 (*b*). Form the appropriate result in *S.R.*
4 (*a*). Clear accumulator.
4 (*b*). Transfer the contents of *S.R.* into the accumulator.

Consider now the mechanism which is controlled by the function part of an instruction.

When an instruction is brought from the store the 5 function digits are routed to the 5 corresponding trigger circuits of the instruction register (see Fig. 10/3). These trigger circuits hold the

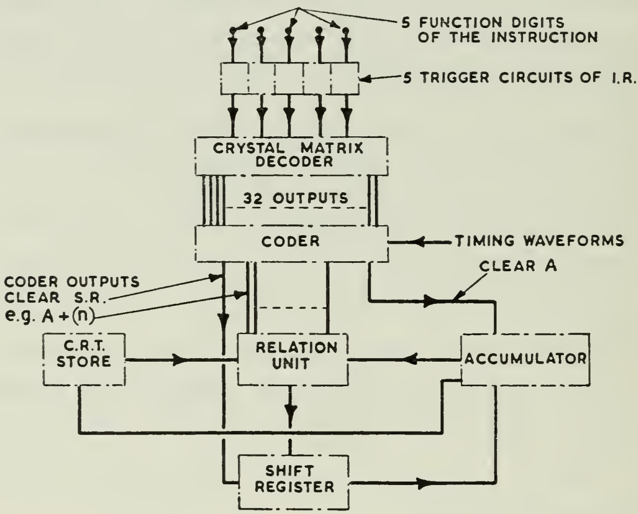


FIG. 10/3. Block schematic diagram of control for performing a function

digits in the form of d.c. levels until cleared by a pre-determined timing pulse before the arrival of the next instruction.

The output from these triggers is taken to a decoding or tree-out circuit consisting of a crystal diode matrix which has 32 output wires. The properties of this matrix are such that a given configuration of input digits energizes one particular output wire, so that a given function is related to a particular output wire.

Many different functions involve the operation of the same gates, e.g. in the four moves described above for carrying out arithmetical and logical operations, the first, third and fourth are common to all instructions of this type. It is necessary therefore to recode the output wires from the decoder. This is done on a set of crystals which is called an encoding or tree-in circuit.

In the case of the conditional instruction (No. 10) it is first routed to *I.R.* in the usual way, but if the content of the accumulator is negative the contents of the address section of *I.R.* are transferred to *C.R.* This means that *C.R.* now contains the address section of the conditional instruction and as this is the address of the required instruction, the latter is the next one taken from the store. The machine afterwards takes the instructions in sequence from this point. If the condition is not fulfilled, *C.R.* is allowed to continue its original sequence without interruption.

The unconditional instruction No. 11 is carried out similarly.

The Relation Unit

In any computer there is a section which will perform mathematical or logical operations, according to the instructions it receives. In the T.R.E. computer this section is known as the *relation unit.* This unit must accept the digital information contained in an accumulator and a C.R.T. store, and be capable of performing any one of eight different operations, and then give a result which may be stored temporarily in an electronic register (the shift register). When carrying out addition or subtraction a third piece of information is provided for any given digital section of the relation unit, that of the *Carry-in* digit, and the unit must provide a corresponding *Carry-out* digit to pass on to the next more significant place of the relation unit.

The eight processes involved* are those of: (*a*) Add, (*b*) Subtract, (*c*) Logical operation *and*, (*d*) Logical operation *not identical*, (*e*) Logical operation *and/or*, (*f*) Transfer of contents of accumulator to shift register, (*g*) Transfer of the contents of address *n* of internal store to shift register, and (*h*) transfer of *minus* the contents of address *n* of the internal store to the shift register.

Representation of Digits and Signs. The T.R.E. computer has been developed as far as possible on the principle that each digit whether 0 or 1 must be represented by a signal, the absence of which, at the appropriate instant, is to represent a *fault*, and this condition must be made known to the operator. Positive pulses are used to represent both 0 and 1, but are fed along two different wires.

The logical operation *and* provides that the result is true if both, and only if both, of two given conditions are true; in terms of digits, the result is a 1 if both of two inputs are each 1. Any other combination of the two input digits must provide the result 0.

* See previous remarks about terminology as compared with that of the Manchester Machine.

This result may conveniently be written in matrix form as—

B \ A	0	1
0	0	0
1	0	1

Turning now to electronics, for the relation *and* a triode valve is arranged as a cathode follower. If one positive input signal is applied to its grid through a large series resistance, and a second positive input signal is simultaneously applied to its anode, then the cathode will give a positive signal.

The logical element *not* is obtained in a two-wire system by interchanging the 0 and 1 wires as in Fig. 10/4.

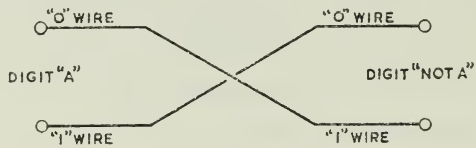


FIG. 10/4. Logical element *Not* or *Complement of*

It can be shown that a network of such units may be so arranged as to provide the results of all the various operations which have been enumerated above.

For an addition circuit it is necessary to provide two systems, to produce the *sum* digit and the *carry-out* digit from two input digits and a *carry-in* digit. The derivation of the *sum* digit will be considered first.

In Fig. 10/5 the squares represent *and* units. Ten of these units are interconnected in such a manner that four sets of input signals are required to provide a result. These are the three pairs of digit signals from the accumulator, store and carry-in and the fourth input signal which may be termed the *Instruction* to carry out the process.

Either a 0 or a 1 exists in the accumulator, and upon the receipt of a positive instruction signal at the two *and* units 1 and 2, either one of the two lines x will become positive according as the accumulator contains a 0 or a 1. This positive signal in turn arrives at the *and* units 5 and 6, or units 7 and 8, so that one of the pair of lines y will become positive, so long as a 0 or a 1 is being applied to the units 5 to 8 from the store.

Similarly, one of the output lines z will be made positive by means of the inputs y, and *carry* 0 and *carry* 1 to *and* units 9 to 12.

A similar network may be drawn for the provision of the *carry-out* digit. Fig. 10/6 shows this arrangement, and comparison of the two Figs. 10/5 and 10/6 reveals the fact that there is much in common. It is therefore possible to combine these two circuits into one, as shown in Fig. 10/7. The crystal diodes form conditional connexions, in order to avoid undesirable cross connexions between the two pairs of outputs.

Additions to the composite circuit of Fig. 10/7 make it possible to

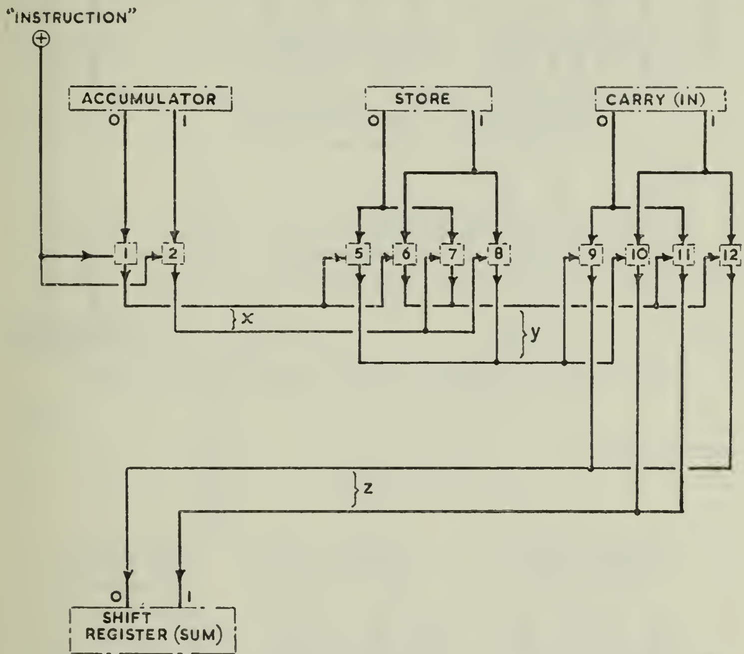


Fig. 10/5. Block diagram of *Sum* section of addition circuit

perform other arithmetical and logical operations, some parts of the original diagram being common to these added processes.

The Pilot Machine

The equipment which has been assembled up to the present time is being used for the purpose of testing circuits designed specifically for a twenty-four-digit, parallel-operated, high-speed computer, and to gain some information regarding the useful life of components under the working conditions of such a computer.

The total time required to read an instruction (to add, say) and to carry out the addition is 40 μsec. A single address code is used.

One C.R.T. electrostatic store of 256 digits is used in the present

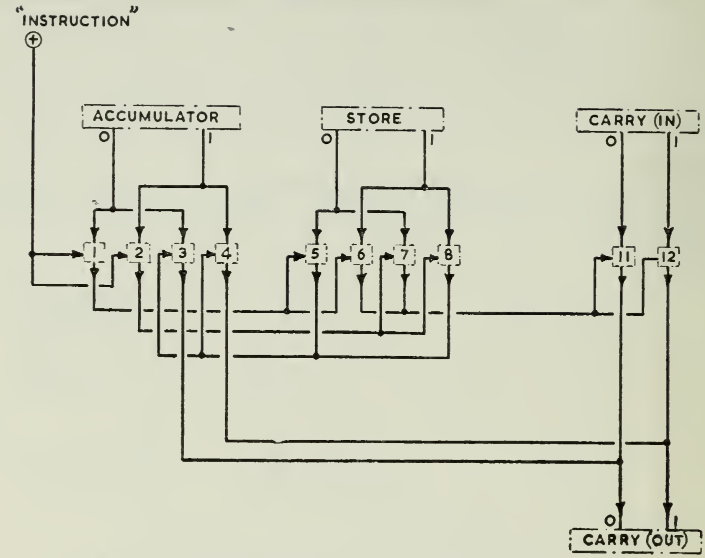


Fig. 10/6. Block diagram of *Carry* section of addition circuit

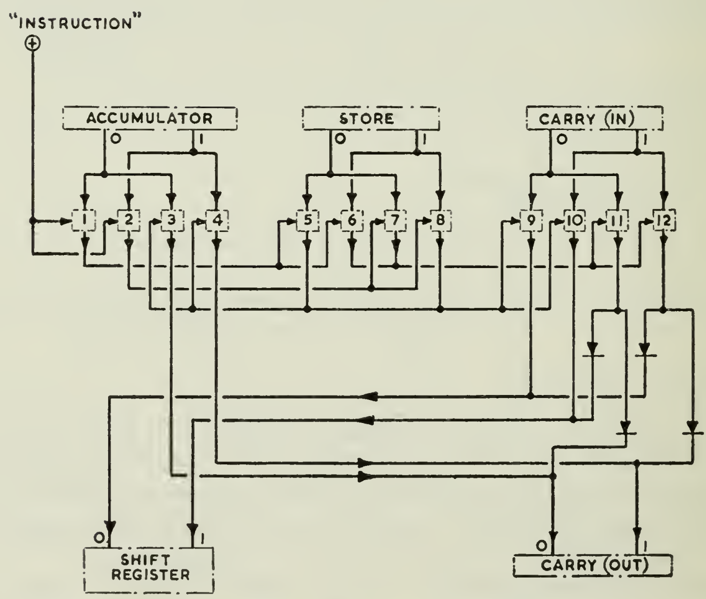


Fig. 10/7. Block diagram of complete addition circuit assembly, and this store is continuously monitored by means of a second C.R.T. (The complete machine will have a capacity of 512 digits per C.R.T.)

In the complete machine the information in *I.R.* arrives from 14 C.R.T. stores simultaneously; this is not possible in a one-digit machine, so that *I.R.* is made a thirteen-stage scale-of-two counter. It follows that a cycle of 32 instructions will be performed upon the

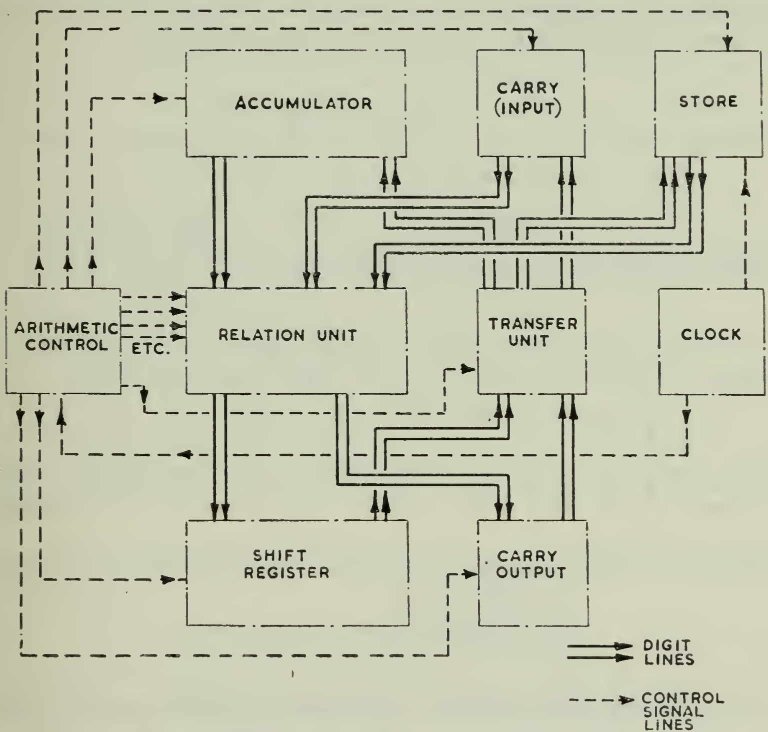


Fig. 10/8. Block diagram of single-digit assembly

contents of a given address and the present contents of the accumulator. The cycle will then be repeated but with the store address increased by unity.

A second result of making *I.R.* a counter is that beats 1 and 2, during which an instruction is found, are not required, so that the one-digit machine continually repeats beats 3 and 4.

To the store are connected an assembly of four registers, a relation unit, a transfer unit, and a control system, the whole being supplied with control pulses from the store (see Fig. 10/8). The store is driven from an electronic oscillator, but in the final machine, the primary source of timing voltage waveforms will be a phonic wheel rigidly fixed to the outer (magnetic drum) store.

The four register units serve the functions of accumulator, shift register, carry-in register, and carry-out register. The last two registers are necessary in the one-digit machine because the carry digit has to be stored from one operation to the next; but in the full-size computer only one register is required, and its function is to "normalize" the signal levels of the carry digit as it is propagated from the least significant to the most significant digits.

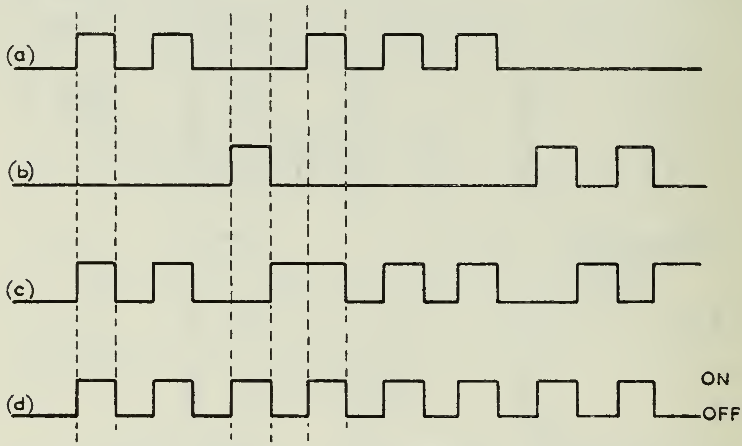


FIG. 10/9. Typical composite-digit waveform used with magnetic-drum store

(*a*) waveform on 1 line
(*b*) waveform on 0 line
(*c*) composite waveform fed to the writing head
(*d*) strobe/clock waveform

Each register unit consists of one pair of trigger circuits with the associated clearing, re-setting, and output valves—one set for 0 and the other for 1. The relation unit performs all the functions required of the final computer. The transfer unit controls moves out of *S.R.*

LIFE AND RELIABILITY TESTS

At first the machine was tested in two parts separately, the C.R.T. store, and the remainder of the equipment which will be called the register assembly.

Register Assembly Tests. Because at first a magnetic-relay counter system was used to record the completed cycles of operation, master control pulses occurred at 50 c/s and a period of time was allowed for the relays to operate before the next cycle was started.

One fault would stop the counter until an observer had noted the conditions and reset the cycle of operations by hand.

Life tests were carried out over a period of more than 1,000 hours. During this time (after initial failures and faults had been cleared) no crystal failures were encountered, but a particular variety of resistor was found to be very poor, and all resistors of this pattern were replaced by others of different manufacture; these then gave no further trouble. Of about 130 valves in the equipment, only 14 were replaced during the test, and of these, eight had been in use previous to the test (in the binary counter) for about 5,000 hours; so that six valves failed within 1,000 hours. Of these, one developed a heater-cathode short circuit, and five lost emission, one after 250 hours, two after 800 hours and two at 1,000 hours.

Continuous periods of faultless operation, without the use of the C.R.T. store, but with d.c. potentials in lieu, of over 90 hours were obtained. This involved over nine million mathematical operations, each of four electronic moves.

Test of Complete One-digit Assembly. The next test was to replace the d.c. supply used as a store by the C.R.T. store as a source of digits, and the relay counter by a completely electronic one. This made full-speed operation possible. Series of computations lasting about three hours have been faultlessly carried out at this high speed of operation.

The Development of a Magnetic Storage Drum for Digital Computers

After preliminary experimental work had been carried out by A. Tutchings[4] a magnetic storage drum was built for use with the proposed twenty-four digit machine. Storage capacity was 2,048 words of 28 binary digits on a drum 4 in. in diameter by 1 in. wide. A second model has been built having a word length of 32 digits and using the same drum dimensions. Both models have a phonic wheel with 2,048 teeth to provide a basic reference frequency.

It is intended to use the reference frequency provided by either the phonic wheel or by a master track to time the operation of the computer which has been designed with aperiodic circuits. This method avoids the inverse problem of synchronizing a drum to an electronic oscillator.

The Drum. The maximum concentration of information in a magnetic surface is obtained with the highest obtainable ratio of coercivity to remanence; for this reason a dispersed coating of magnetic iron oxide has been used which allows a cell of length 0·006 in. with only 2 db loss of response. So that information can be obtained at the designed 20 μsec intervals, a peripheral velocity is required of approximately 25 ft/sec. A drum 4 in. in diameter rotating at 1,500 r.p.m. meets this figure and gives a capacity of 2,048 words.

Magnetic Heads. A single magnetic head performs the function of reading and writing on any one track and consists of three toroidal laminations of metal 0·005 in. thick, wound with 50 turns of wire; a magnetic gap of 0·001 in. is maintained by a non-magnetic spacer, and only the central lamination is used at the gap, the other two being cut back a little. The head is mounted in a brass holder about 1 in. square by $\frac{1}{8}$ in. thick.

The required writing current is 250 mA and the read output is approximately 200 μV at an impedance of about 6 ohms.

Interference. 1. The width of any particular *track* is only slightly greater than that of the writing head producing it. A separation between adjacent tracks of half the minimum wave length is needed to avoid pick-up from an adjacent track. As a result the maximum number of tracks is 100 per inch and the digit density is therefore over 16,000 per in.[2] Under these conditions pick-up between any head and an adjacent track is less than — 40 db.

2. For two adjacent *heads* in the reading condition stray flux from one could be picked up by the other, but with a spacing of $\frac{1}{8}$ in. between heads this type of interference is found to be 30 db below signal level, which is satisfactory.

Acknowledgments

The authors wish to thank the Chief Scientist of the Ministry of Supply, Great Britain, and to the Controller, H.M. Stationery Office for permission to publish this paper. They also wish to record the contributions of G. H. Perry on Parallel C.R.T. Storage, of G. Davies on design of the Control Organ, and of E. Hesketh on Magnetic Storage and Basic Waveforms.

REFERENCES

1. Williams, F. C. *Proc. Roy. Soc.* "A" **195**, 1042 (Dec., 1948) 265.
2. Williams, F. C. and Kilburn, T. *Proc.I.E.E.* **96**, Pt. III, 40 (Mar., 1949) 81.
3. Perry, G. H. *Nature* **168** (Sept., 1951) 372.
4. *Report of a conference on high-speed automatic calculating machines.* Cambridge (June, 1949).

Chapter 11

THE IMPERIAL COLLEGE COMPUTING ENGINE

THIS IS A SMALL RELAY MACHINE of parallel type, working in the binary scale to an accuracy of 20 digits (including a sign). This corresponds to over five decimal digits.

Input to the machine is in the form of signed five decimal digit numbers, which are punched on cards and passed through the feed of an ordinary Hollerith tabulator. A relay mechanism is provided for interpreting the signals from the tabulator, whereby the automatic conversion from decimal to binary form is accomplished with only four additions, which are carried out on the main arithmetic unit of the machine. It has been found possible to arrange that this conversion can take place at the normal rate of card feed; namely about 50–60 numbers per minute. Input is arranged in blocks, the last card in each block having a special punching to terminate card feed.

Output from the machine is also in the form of signed five decimal digit numbers, which are printed on the print bank of the tabulator in one of five columns, and in one of five positions in the chosen column. These positionings can be adjusted by the machine in the course of its calculations.

The conversion from binary to decimal form is performed by the main arithmetic unit. This multiplies the number by ten, removes and stores the leading four digits, and repeats this cycle a further four times. The sets of digits are the binary forms of the successive decimal digits, and the stores used to hold them are connected to the Hollerith printing mechanism to control the digits printed. Two other stores, each of three binary digits, control the positioning of the number to be printed. Another single-digit store controls the paper feed.

When all the stores are set, the tabulator starts a printing cycle. The machine can start its next operation as soon as the tabulator is started, and does not have to wait till the end of the cycle. The duration of the conversion process is about two seconds.

Several forms of storage are used in the machine. In addition to storage for the normal numbers used for calculation, there are stores for small numbers (5 digits plus sign) used for counting or recording exponents, and similar ancillary operations. There is also separate storage for instructions, for reasons explained later.

There are twelve relay stores for each type of number. These consist of a high-speed relay for each binary digit, wired so as to remain energized after a first energizing pulse. The store is cleared by energizing an additional relay. A group of multi-contact relays acts as a gate to connect the store to the highway, the main trunk for the passage of numbers within the machine. The time to pass a number into or out of these stores is about 0·1 to 0·2 sec.

The main storage for numbers is on punched tape. There are four units, each of which records a twenty-digit number by punching two rows of holes on tape. This may be read later by means of a set of brushes. The electrical circuit is so arranged that the time taken to record a number is no greater than with a relay store, the punching being performed while the machine proceeds to its next operation. Reading from the tape was a little slower, since the tape had to be moved through the width of one row to read both rows of the number in succession. This took about 0·25 to 0·5 sec. This has now been improved so that both rows are read together. Reading now takes no longer than punching. It is possible to arrange to copy automatically from one tape to another, as the former is read.

To eliminate the use of expensive storage for constantly occurring numbers, about 80 numbers, such as π, e, the simple binary fractions, and numbers of the form 2^{-P} $(P > 0)$ are permanently available and do not need to be fed into the machine at all. In addition, there are five pre-set stores for constants which arise in individual problems.

The form of instructions for this machine is rather novel. They are stored as five-digit words or numbers. A complete instruction consists of a group of these words fed to the machine in order. The first word specifies the operation or type of operation to be performed; the following word completes the specification of the operation, if this is required, and then follow the address(es) of the store(s) containing the number(s) required for the operation, and finally, if it is needed, the address of the store which is to be used to store the result. This method of control has the advantage that if the result of a calculation is one of the operands in the next operation, it need not be stored, but can be left in the arithmetic unit. This saves the time of transport of the number into and out of the arithmetic unit, and cuts down the amount of storage required for instructions. Thus each of the basic operations of the machine has four variants.

Since high-speed storage in this machine is rather expensive, the storage of instructions is in the form of holes in paper tape, and so they are not alterable during the calculation. Because calculation consists of certain cycles of operations repeated time after time on different numbers, means must be provided for obtaining different numbers from the same addresses (since the addresses are as unalterable as the operations). Three main means of achieving these ends have been incorporated in the machine.

The tape stores are used as vector stores,* each successive number from the corresponding address being the next element in a vector.

The *tree* address consists of a *tree* associated with a short store, so that the actual address used, if the tree address is called, is that in the store. The address is thus alterable by performing arithmetic on the contents of the store. The third method of altering addresses is known as the cyclic system of stores. These stores are used in the usual manner, but a special signal gives the labelling of the stores a cyclic permutation. This device is useful for problems involving differencing and the use of recurrence relations.

Although it is essential to have some means of discrimination in a machine, if it is to be able to control the course of the calculation without human intervention, only a small proportion of the instructions in a calculation are concerned with discrimination. The bulk of the instructions follow one another in a fixed and pre-determined order. The complete set of instructions naturally breaks up into sequences of instructions, each ending with a discrimination. These sequences form the natural units for storage in the machine, and the tape corresponding to each sequence is wound on a drum. These drums are rotated by ratchet motors to enable the successive words of the sequence to pass to the control of the machine through a set of sensing brushes. Means are provided to switch from one drum to another at the end of each sequence, and to return the discarded drum to its proper position for re-use.

There are two arithmetic units in the machine to deal with long and short numbers respectively. The provision of a separate unit to deal with short numbers obviates the necessity for shifting numbers from the main unit merely to perform counting operations. This unit can add or subtract two numbers, count up or down from a number, and change the sign of the result, if required. Multiplication or division is rarely required for this type of number and is not provided.

* For our purposes, any set of numbers with a fixed ordering may be considered as a vector.

The main arithmetic unit deals with twenty-digit numbers, and can add, subtract, multiply or divide two numbers. It can also be used to shift a number in either direction. The number of places shifted can be specified by the type of operation, by the contents of a small store, or by the number of leading insignificant digits in the number.

In this unit, numbers are treated as fractions so that the last-mentioned operation shifts the number until its modulus lies between $\frac{1}{2}$ and 1. The auxiliary unit treats short numbers as integers.

Since certain multiples occur frequently, the main unit is arranged to give these multiples of a number without having to store the multiplier, thus saving both storage space and time. Each of these operations is estimated to take about half to one second.

In addition to the arithmetic operations, there is a set of operations which controls the structure of the machine. For example, one such instruction can be used to switch the machine so that products are rounded off before being stored, and later in the calculation another instruction of this type can adjust the machine to leave products unrounded. This type of instruction is also used to alter the copying arrangements of the tape units.

Another set of operations enables a pre-set group of operations to be performed as a single instruction. Printing and input are examples of such instructions.

Although it is possible to arrange that the discriminations necessary in a calculation always depend on the sign of a number, it is more convenient to be able to make these discriminations on a variety of conditions. Eight such conditions are provided in the machine, and provisions are also made which enable the next possible sequences to be fixed in advance, or determined during the calculations.

Various special stores are incorporated which transform in various ways the number sent to them. For instance, one store accepts only positive numbers, negative numbers clearing the store. These special stores can reduce very considerably the difficulties of programming.

The whole machine contains about two thousand relays, and about two dozen uniselectors. Although its operating speed is very slow compared with electronic machines, there are large numbers of calculations for which it offers a considerable saving in time and energy over hand computing, and being more robust, it will probably be more trouble-free than some electronic machines.

Chapter 12

THE ROYAL AIRCRAFT ESTABLISHMENT SEQUENCE-CONTROLLED CALCULATOR

Most of the automatic calculating machines described in this book are very fast. They were designed, in the main, to solve complex calculations in which the amount of computing is large compared with both the quantity of information being fed into the machine and the number of answers coming out. For other problems, full advantage cannot be taken of their high computing speed and their performance is limited by the much slower processes of input of data and output of answers. Furthermore, most of these machines operate internally with numbers expressed in the binary scale. This necessitates a double conversion, from decimal to binary before computation and back again afterwards, which swamps the computing proper unless the latter is, again, substantial in relation to the input and output.

Several such machines are clearly needed in this country to handle the big jobs; there remain, however, many problems too long for ordinary desk machines and yet uneconomical for the big fast machines. It is just this intermediate class of problem which predominates in the Royal Aircraft Establishment. Aeronautical research—like much other applied research—involves the reduction of large quantities of experimental data—from wind tunnels, aircraft flight instruments, structural test specimens, etc. The computing processes are often quite simple and an accuracy of more than four figures is seldom warranted; but the mass of data is vast, since expensive research equipment, such as supersonic wind tunnels, demands extensive instrumentation to extract the maximum information from each test.

A small calculator to deal primarily with this class of work is therefore being designed at R.A.E.—largely by Mr. E. J. Petherick—and a prototype built by an industrial firm. Emphasis has been placed on flexibility, simplicity, low cost and ease of use. It differs from most other British machines in that it handles numbers in decimal form throughout and uses what is known as a *floating decimal point*. Each number is expressed in the standard form $\pm p \times 10^j$, where the *significant figures*, p, lie in the range 0 to 1, and the *exponent*, j, may have integral values between -19 and $+29$. Thus the number 147·8264 appears in the machine as $0 \cdot 1478264 \times 10^3$, and $-0 \cdot 002375$ is read as $-0 \cdot 2375 \times 10^{-2}$. The user therefore need keep only a cursory eye on the size of computed quantities to check that they remain within the stated limits. Numerical data can be fed to the machine with up to eight significant figures, but nine are retained throughout the computation to minimize round-off errors. Answers can be displayed to eight figures or forced to any lower accuracy desired.

Both numerical data and operating instructions in coded digital form are fed to the machine on a number of punched tapes, similar to but more robust than the tele-type tapes used on other British machines. A four-hole code defines each decimal digit of a number or instruction, and the tape readers have been designed to reduce to about 10 msec the delay between the arithmetic unit calling for and receiving a number from tape. Hollerith punched cards will provide an alternative form of input.

The main internal store for numbers and instructions is a magnetic drum similar to those used on other British machines. Storage will be provided for nearly 10,000 *words* (numbers or instructions). Operating instructions will normally be transferred to the drum before computing starts; and the numbers used in each operation can be read either from the drum, direct from one of the input tapes, or from punched cards.

Final answers are tabulated by one or more electric typewriters, each of which provides up to twenty carbon copies of an original suitable for photostatic reproduction—an important asset where large numbers of results are required quickly for scrutiny or further analysis.

Alternatively, the final answers can be punched on tape or cards.

A *two address* control code is used, typical instructions being: "Add the number in address X to the accumulator and send the sum to address Y," or: "Form the product of the numbers in addresses X and Y and add the product into the accumulator." An instruction usually consists of eleven decimal digits. The first pair denotes the operation to be performed, while the last eight form two four-digit addresses indicating the source(s) of the number or numbers to be operated on (known as operands) and/or the destination to which the answer is to be sent. The third digit of an instruction, known as the *modifier digit*, provides facilities for modifying orders on similar lines to that provided by the *B*-tube of the Ferranti machines.

In addition to the usual basic arithmetic operations, other instructions control the modification of orders and organize the desired *flow* of the calculation. In an actual computation, control will seldom traverse a simple linear path along a sequence of instructions; it will repeat some groups of instructions and omit others; and the path taken may depend on results already obtained by the machine. While instructions are usually obeyed in sequential order, facilities are provided for transferring control to a new point in the set of orders specifying the complete computation. As in other machines, this control shift may be unconditional or conditional. In the latter case, the machine selects one of two instructions depending on the sign of a number in some specified position in the machine.

The *arithmetic unit* consists mainly of a pulsed network of cold cathode valves: ten state *Dekatrons* and two state trigger tubes. The Dekatron consists of a central anode surrounded by a ring of ten cathodes each separated from its neighbours by guide electrodes. The valve has ten stable states; in any one such state a discharge passes between one cathode and the anode. By suitably pulsing the guide electrodes the discharge can be transferred to successive cathodes at a rate of up to 20,000 steps per second. To see how Dekatrons may be used in decimal computing, let us consider the action of a single valve suitable for storing one decimal digit of a number or order. The ten cathodes, and the corresponding ten stable states, may be thought of as numbered 0 to 9. A digit n in transit within the machine is represented by a sequence of n pulses in one wire. The Dekatron may thus be set (from its zero position) with any digit n by sending n pulses to its guide electrodes; it may be read by feeding ten pulses and counting those arriving after that pulse which causes the discharge to step from cathode 9 to cathode 0; preceding pulses form the complement on ten of that digit (i.e. $10-n$). The Dekatron may be cleared by supplying pulses only until the discharge steps from cathode 9. To add two digits, m and n, the Dekatron is pulsed $m + n$ times; if the discharge steps from the ninth cathode, a carry to some more significant digit is needed.

The *arithmetic unit* comprises an *accumulator* (a ring of Dekatrons) a *register* and a *multiplication table* for dealing with the significant figures of numbers; an *index unit* which handles the corresponding exponents; and a *pulsing unit* which emits groups of pulses obtained from a pulse generator.

Numbers can be transferred from the tapes or magnetic drum to the register, on which each digit is represented by the states of a quintet of trigger tubes, in a code similar to that used on the tapes and drum. Each such digit, n, can be added to the accumulator by pulsing the appropriate Dekatron n times; and all significant digits of a number are added to the accumulator simultaneously, in parallel, during the first nine pulses of a cycle of ten. A second operand can be added to the accumulator in a second cycle, inter-digit carry being arranged during the last pulse of that cycle, and means are available to position two such operands on the accumulator in their correct relation, should their exponents differ.

The Dekatrons can move in one direction only, and so subtraction must be arranged by addition of *nines complements*. Each significant digit n of a number to be subtracted is then represented by feeding $(9-n)$ pulses to the appropriate Dekatron in the accumulator. A small adjustment known as end-around carry may have to be made, thus—

Problem	837	*Machine Method*	837	
	− 243		756	(nines complement)
	594		(1)593	(Add)
			1	(End-around carry)
			594	

Multiplication involves, besides the accumulator and the register, a built-in *multiplication table*, which enables the multiplicand, or any multiple of it up to the fifth, to be added to or subtracted from the accumulator. The multiplicand is held on the register and the multiplication table is controlled by successive pairs of digits of the multiplier. Multiples above the fifth are not required, because short-cut multiplication—familiar to users of desk calculators—is used. A multiplier 157, for example, is regarded as—

$$\begin{aligned} & 100 + 50 + (10 - 3) \\ = {} & 100 + (50 + 10) - 3 \\ = {} & 100 + (100 - 40) - 3 \\ = {} & 200 - 40 - 3, \end{aligned}$$

and so the second multiple of the multiplicand would be added to the accumulator and the fourth and third multiples subtracted in their correct relation one to the other, to form the complete product. Several arithmetic properties of the decimal multiplication table can be utilized to minimize the equipment required for multiple formation.

The accumulator can form the sign and first nine significant figures of the sum of numbers or multiples fed to it in succession in the form of a train of pulses. When such a sum is to be stored or displayed, it is first transferred to the magnetic drum before being fed out to the tape or card punches, or typewriters.

The speed of the machine is set by the time of revolution of the magnetic drum—about 10 msec. Addition and subtraction operations take 10 msec, while multiplication of 2 eight-digit numbers requires 20 msec. As mentioned previously, a number on a tape can be obtained "on call" in 10 msec, i.e. in the time required for one drum turn.

Finally, it should be emphasized again that this small machine is intended primarily for the reduction of experimental data. To be used to full advantage, it must be associated with various ancillary devices, particularly for recording experimental data in a form which can be read directly by the machine. For this class of work, the arithmetic calculations are by no means the whole story; the situation demands a co-ordinated system of instrumentation and computing equipment and cannot be met by either standing alone.

ACKNOWLEDGEMENTS

Thanks are due to the Chief Scientist, Ministry of Supply, for permission to publish.

Chapter 13

CALCULATING MACHINES AT THE BIRKBECK COLLEGE COMPUTATION LABORATORY

SINCE 1947 THIS LABORATORY has been instrumental in the development and construction of three main types of general purpose computing machine—

(*a*) A.R.C. Automatic relay calculator.
(*b*) S.E.C. Simple electronic computer.
(*c*) A.P.E.(X).C. All purpose electronic *X* computer.

The whole purpose of the work has been to build a small, reliable automatic computer having a performance comparable to that of the numerous large machines at present under construction.

AUTOMATIC RELAY CALCULATOR

This is a prototype machine built for the British Rubber Producer's Research Association; it has parallel operation arithmetic and control units constructed almost exclusively of Siemens high-speed relays—almost 800 in number. In its original form it had a drum-type magnetic store of capacity 256 twenty-one-digit binary numbers, but it has since been considerably reduced in size by the provision of an adequately fast electromechanical store in place of the original drum.

Input and output are via punched-paper tape and standard Creed teletype equipment. Operating speeds are 0·02 sec for addition and subtraction, and 1 sec (in the worst case) for multiplication and division.

It is interesting to note that the original magnetic-drum storage of A.R.C., was the first of its type to be constructed anywhere, and has been copied extensively by most other groups.

SIMPLE ELECTRONIC COMPUTER

This was constructed as a prototype of an all-electronic, high-speed machine. It employs magnetic-drum storage for 256 twenty-one-digit binary numbers, having a digit repetition rate of 12·5 kc/s.

The arithmetic unit is of the serial type and the control serio/parallel. In view of the experimental nature of the machine no multiplier was constructed. Addition and subtraction take 1,000 μsec.

The control is of the two-address type—the next-order location being specified in the current instruction. This important feature makes possible *optimum coding** and the overall speed of the machine in a variety of long computations is only about 6 per cent less than would occur if the inter-word access time was instantaneous.

Input is via punched tape read photo-electrically; unlike most other readers, however, this uses only one photocell and data are emitted serially.

Output is either to a reperforating punch or to teleprinter.

All-purpose Electronic (*X*) Computer

This group of machines (the (*X*) indicating the particular group for which a machine is to be built) has, so far, four members—

A.P.E.X.C. Birkbeck College.
A.P.E.R.C. British Rayon Research Association.
A.P.E.H.C. British Tabulating Machine Company.
A.P.E.N.C. Norwegian Board for Computing Machines.

The machines are properly engineered versions of S.E.C. built with miniature components. The digit repetition rate is 50 kc/s. Storage capacity is of the magnetic drum type and gives 512 thirty-two-digit binary numbers. The machine is provided with a high-speed, short-cutting multiplier and takes 500 μsec for addition and subtraction, and $(n \times 500)$ μsec for multiplication, where n is the number of "live" digits in the multiplier.

Only 415 valves are used, including all diode elements. Power consumption is 1·5 kW.

Input and output equipment varies with the requirements of the users, three types being available—

1. Punched-paper tape read photo-electrically as input, output to reperforator or teleprinter.

2. Magnetic-tape input and high-speed output. Printing via a teleprinter.

* Optimum coding implies that each instruction is extracted during the arithmetic operation which precedes it, and assumes that numbers which will be needed in the computation are available as soon as they are required (see page 110 where this point has been further discussed).

3. Punched-card input and intermediate store. Output via Hollerith tabulator.

By virtue of their two-address code and optimum programming, machines of A.P.E.(X).C. type are little slower in operation than some of the larger machines at present under construction.

Chapter 14

COMPUTERS IN AMERICA

ALTHOUGH THIS BOOK is intended to describe the development of computing machinery in England it would be incomplete without a brief account of progress in America. The first modern computers were built there, and more machines are working in the U.S.A. than exist in the whole of the rest of the world. Unfortunately we have not been able to include first-hand accounts of any of the American machines, and so material for this chapter has been extracted from published descriptions to which the reader is referred for details.

THE HARVARD MARK I CALCULATOR

This machine was invented by Professor Aiken of Harvard University, and B. M. Durfee, F. E. Hamilton and C. D. Lake of the International Business Machines Corporation. Its design was begun in 1939 and it was put into service in 1944 at Harvard where it has been ever since. It is the first machine actually to be built which exploits the principles of the analytical engine as they were conceived by Babbage a hundred years before.

The machine consists of a 51 ft panel 8 ft high on which are mounted the tape readers and the relays and rotary switches which control the machine. Almost all the operations of the machine are controlled by mechanical switches of one kind or another, many of which are driven from shafts turned by a four horse-power motor. Most of the components from which the machine is made were standard I.B.M. products.

The machine handles numbers in decimal form and works to 23 significant figures. The operations of the machine are controlled by twenty-four-hole punched tape which advances at about 200 steps per minute. This means that the basic operation time (the addition time) is about 0·3 sec. Multiplication and division are done by using a built-in multiplication table; the computation of logarithms or of sines and cosines takes about a minute for each value, using special routines and a series of standard values which are built into the logarithmic unit and the sine unit. Numbers are stored in relay units, on tapes, and on a series of hand switches which can be set up for each computation. The machine can read from any one of its input tapes, and can transfer control automatically to the main sequence tape or to any one of a series of subsidiary tapes which contain subroutines.

This machine has been in continuous use ever since it was built. At first it computed ballistics tables and other special tables for the American Government, but since the end of the war it has spent the greater part of its life computing mathematical tables of all kinds. Harvard Mark I cannot fully exploit the power of automatic judgment which Babbage proposed to incorporate in his analytical engine. When first it was built, the only alternatives between which the machine could choose were to "continue with the next instruction on the tape" or "stop" and it decided between these two according to whether the number it had computed was greater or less than the amount specified by the programmer. The flexibility of control has since been improved, but for the first part of its life it could not compute its own programme in the manner which Lady Lovelace suggested.

The Electronic Numerical Integrator and Calculator

This was the first all-electronic computer to be built. It was planned by Dr. J. W. Mauchly and Dr. J. P. Eckert at the Moore School of Electrical Engineering of the University of Pennsylvania for the Ballistics Research Laboratory at Aberdeen Proving Ground. It was completed in 1946. It contains 18,000 valves and 1,500 relays and is probably the largest machine that will ever be built. In many ways it is an electronic analogue of the Harvard Mark I machine, but it works several hundred times faster. The large size of the machine is due to the fact that the whole of the high-speed storage is in valve staticisors, and to the relative inefficiency of valve circuits which work in a decimal scale. The machine has been modified once or twice since it was first built but it has been in almost continuous use since 1946.

The International Business Machines Corporation Selective-sequence Electronic Calculator

This machine was completed and put into operation by I.B.M. at their Headquarters in New York early in 1948. It could be seen from the street by passing pedestrians who affectionately christened it "Poppa." It was a very large machine and contained 23,000 relays and 13,000 valves. All arithmetic operations were carried out by the valves and so it was more than 100 times as fast as the Harvard Mark I machine. It had three types of memory; a relatively small high-speed store in valves, a larger capacity store on relays, and an indefinitely large store on eighty-column paper tape which could be used either in loops or if need be in great coils which could be removed from the machine and replaced at a later stage in a calculation. Instructions and input data were punched on tape and there were 66 input reading heads so arranged that the control of the machine could be transferred automatically from one to another, as the calculation proceeded. It was probably the first machine to have a conditional transfer of control instruction in the sense that Babbage and Lady Lovelace recommended.

The machine was able to undertake calculations on a commercial basis and was probably the first machine of this type to make a profit in this way.

During the last year or two of its life it worked for the U.S. Government, undertaking one computation for the Atomic Energy Commission which occupied it for six months and involved the storage on tape and subsequent use of many millions of digits. It would have taken more than 100 years to do the work by hand. Before it was monopolized by Government work it computed lenses, analysed the geology of oil wells, did many calculations for astronomers (see page 285) and proved itself to be a most versatile and reliable instrument. Most of the mistakes which have been found in calculations have been due to errors on the part of the mathematicians who propounded the problems in the first place.

The machine in operation must have been the most spectacular in the world. Thousands of neon lamps flashed on and off; relays and switches buzzed away and the tape readers and punches worked continuously. The machine was dismantled in August, 1952.

The Bureau of Standards Eastern Automatic Computer

This machine is installed in the Bureau of Standards in Washington, where it has been working for 24 hours a day since 1950. The engineers have contrived to modify it, at the same time as it has been in use by the Bureau, for all kinds of mathematical problems. Originally it was similar in its basic design to the E.D.S.A.C. and the A.C.E. (see page 135), in that its main memory was in mercury delay lines. It differs from these machines, however, in that it is all A.C. coupled, and makes extensive use of small pulse transformers in order to reduce the impedance of long coupling leads and to minimize cross coupling and the effect of stray capacities, which are a prolific source of trouble in many machines. This mode of operation involves the use of large numbers of diodes—and no less than 18,000 germanium crystals are in use. During the past year, a Williams memory has been added to the machine—the combination of delay lines and cathode-ray-tube stores is to be found in no other machine. The memory tubes are worked in parallel but the data is "serialized" and used in the same way as the pulse chains emerging from the mercury tanks. Input and output are via magnetic tape, which serves as an auxiliary memory.

The machine has been overwhelmed with problems ever since it was first put into service, and it has done many calculations for both the Bureau of Standards and the Atomic Energy Commission. It is hoped to devote it to an analysis of the organization, or perhaps one should say the "Econometrics," of the American Army Air Force and ultimately perhaps of the whole economic structure of the country, in the manner which is described in Chapters 21 and 23. This enterprise is known as "Project Scoop."

The Machine of the Institute for Advanced Studies, Princeton

In 1945, Professor J. von Neumann, who was then working at the Moore School of Engineering in Philadelphia, where the E.N.I.A.C. had been built, issued on behalf of a group of his co-workers a report on the logical design of digital computers. The report contained a fairly detailed proposal for the design of the machine which has since become known as the E.D.V.A.C. (electronic discrete variable automatic computer). This machine has only recently been completed in America, but the von Neumann report inspired the construction of the E.D.S.A.C. (electronic delay-storage automatic calculator) in Cambridge (see page 130).

In 1947, Burks, Goldstine and von Neumann published another report which outlined the design of another type of machine (a parallel machine this time) which should be exceedingly fast, capable perhaps of 20,000 operations per second. They pointed out that the outstanding problem in constructing such a machine was in the development of a suitable memory, all the contents of which were instantaneously accessible, and at first they suggested the use of a special tube—called the *Selectron*, which had been invented by the Princeton Laboratories of the R.C.A. These tubes were expensive and difficult to make, so von Neumann subsequently decided to build a machine based on the Williams memory. This machine, which was completed in June, 1952 in Princeton has become popularly known as the Maniac. The design of this machine has inspired that of half a dozen or more machines which are now being built in America, all of which are known affectionately as "Johniacs."

THE WHIRLWIND, MASSACHUSETTS INSTITUTE OF TECHNOLOGY

This is among the largest machines ever built, and it is probably the fastest now working. It performs about 20,000 single-address operations per second on sixteen-digit words. It exploits a special electrostatic store which was invented in the laboratory. Sixteen tubes each contain 256 binary digits. This memory is too small for most problems, but it will shortly be increased threefold.

The machine is so fast that it is hoped to be able to use it for solving problems in "real time," that is to say it should be able to keep up with the motion of an airplane by solving the equations of motion continuously and making use of telemetered data on the position of the controls and of the motion of the plane. Studies of this kind will help to solve many of the problems of aircraft stability and of high-speed flight.

THE I.B.M. CARD-PROGRAMMED CALCULATOR

For many years the International Business Machines Corporation has built complicated accounting machinery which is based upon the use of punched cards. By combining several of their units, one of which is a recently introduced multiplying punch, together with one or more specially built high-speed memories, each of which will hold 16 ten-digit numbers, their engineers have been able to produce a very flexible universal digital computer which, since its programmes are punched on standard cards, has become known as the C.P.C. Several dozens of these machines have been built and delivered, and in spite of the fact that their speed is less than that of most of the machines which have been described in this book, it is probably true to say that so many are in use that they have done between them an amount of computation which is comparable to that done on all the other computers which have been made. The limitations of this machine are due to the relatively small size of its memory, and to the small number of subroutines which can be plugged up on it at one time. The connexions which produce these subroutines cannot easily be modified once they have been set up, since they involve the use of a board which is rather like a small manual telephone exchange. In spite of this they have proved useful, fast and versatile, and many engineering firms have found them to be invaluable in their calculations. The reader will be interested to read Lady Lovelace's criticism (see pages 378–95) of this type of machine.

The U.N.I.V.A.C.

This is a very large and complicated machine which was designed by Eckert and Mauchly, who built the E.N.I.A.C. during the war. It was built by the Remington Rand Corporation, and it is, so far as we know, the only large-scale computing machine which has been built in America for commercial use. Several of these machines have been commissioned, and one has been in use for a year or so by the Bureau of the Census.

The machine consists of a central computing element, which uses mercury delay lines and is in many ways similar to the E.D.S.A.C., although it works at a much greater speed (the pulses move through the tanks at a p.r.f. of 2·5 Mc/s, as compared to the 0·5 Mc/s of the older machine). A most important feature of the machine is the very large backing-up store on large metal magnetic tapes, which will hold some hundreds of millions of digits. The tapes are driven by mechanisms called *uniservos*, which will accelerate the tape and control it precisely under instructions derived from the central computer which can therefore be used not merely to handle very long and complicated computations, but to perform the commercially important work which may best be described as *data processing*.

Information is written on to the magnetic tapes from manually operated typewriter-like keyboards, and the output of the machine, which comes on to the tapes, can be printed by standard typewriters, suitably modified. This means of course that the processes of preparing input data and of printing the final answers can be done quite independently of the main machine.

It is probable that several of these machines will be in use before long, and it will be most interesting to observe their impact on American business.

Other Machines

Machines of many types have been and are being built in other Universities and Government establishments and by several large commercial firms. Progress is continuous.

PART THREE

APPLICATIONS OF ELECTRONIC COMPUTING MACHINES

Chapter 15

MACHINES FOR THE SOLUTION OF LOGICAL PROBLEMS

With this we complete the overthrow of speculative philosophy—
A. J. Ayer

The origins of systematic logical method are of great antiquity. It was, however, the publication in 1847 of *The Mathematical Analysis of Logic* by George Boole which laid the foundations of the modern science of symbolic logic.

It is clear even now that Boole's work aroused very great interest. An indication of the thoughts of this period is given in a paper published only four years later—in 1851—by the Reverend Alfred Smee, Surgeon to the Bank of England. Mr. Smee's work bore the imposing title, *The Process of Thought adapted to Words and Language, together with a Description of the Relational and Differential Machines.* These machines were described in outline, and were of a most ambitious character, one of them being intended to occupy an area about the size of London; not unnaturally, they were never constructed.

In 1854 Boole published another work, *Of the Laws of Thought*; after this there appears a gap in the literature for about ten years. In 1864 there appeared the first important work of W. S. Jevons, entitled *Pure Logic*, and in 1866 Jevons read a paper before the Manchester Literary and Philosophical Society (*Proceedings*, (3rd April, 1866) 161) in which the idea of the *solution of logical problems by the method of elimination of classes* was clearly expounded.

The Method of Elimination

The importance of this concept is sufficient to justify its closer examination. The following statement will serve as an example: "Only members or their guests may play over the Blankshire Golf Club's course."

This statement, reduced to the form in which it will be used, contains three distinct *components*; by a component is meant a property or status which divides the universe into two classes characterized respectively by possession and non-possession of that status. The components here are; firstly, membership of the Club (any given person either is or is not a member); secondly, the status of members' guest; thirdly, eligibility to play over the course. Let it now be supposed that, although the components are known, the actual statement is for the moment unknown. Then since there are three components, and each divides the universe into two classes, there may exist in all $2 \times 2 \times 2 = 8$ classes. It is easier to enumerate these if a symbolic notation is adopted; for example, let *A* denote eligibility to play, *a* ("not *A*") ineligibility, *B* membership, *b* non-membership, *C* the status of members' guest, *c* the opposite. The eight classes may then be written

ABC, *aBC*, *AbC*, *abC*, *ABc*, *aBc*, *Abc*, *abc*

Before the statement is made, nothing is known about the existence or non-existence of any of these classes. *The making of the statement is an assertion of the non-existence of certain of the classes;* this is the method of elimination or exclusion referred to. Referring back to the statement, it may be seen that the second, fourth, sixth and seventh classes are eliminated by it; that is to say, they are classes the existence of which would be inconsistent with the statement.

Truth Tables

It is clear that in more complicated problems it would not be possible to do the elimination intuitively, and it is therefore necessary to consider how it may be done by systematic methods.

Two components only will first be considered. These yield four classes, *AB*, *aB*, *Ab*, *ab*. Most simple phrases used in logical statements may be given exact equivalents in terms of these classes; for example—

(*A* and *B*)	Includes *AB*	Excludes *aB*, *Ab*, *ab*
(*A* or else *B*)	Includes *Ab*, *aB*	Excludes *AB*, *ab*
(*A* or *B*)	Includes *AB*, *aB*, *Ab*	Excludes *ab*
(*A* if and only if *B*)	Includes *AB*, *ab*	Excludes *aB*, *Ab*
(If *A*, then *B*)	Includes *AB aB*, *ab*,	Excludes *Ab*

and so on.

There are in all sixteen relationships possible with two components, although some are quite trivial. Various mathematical methods are available for their manipulation; for example, matrix algebra is convenient. The example above may be dealt with without advanced means by re-writing the statement or "rule" as

A if and only if (*B* or *C*)

Now write D for the "complex" component (B or C). By the above definitions; D includes BC, bC, Bc; d includes bc. Also by the definitions, (A if and only if D) includes AD, ad. Hence the classes included in that defined by the rule are those given by combining A with a class included in D, i.e.—

$$ABC, \quad AbC, \quad ABc$$

together with the only class given by combining a with the class included in d, i.e. abc. These are of course the four non-eliminated classes or "solutions."

The method known as the preparation of *truth tables* consists of a systematic application of this procedure. Relationships between many components may always be dealt with by repeated application of relationships between two (simple or complex) components; hence this method is of general application.

A problem may of course have more than one rule. In this case each rule may be applied successively to those combinations which have not been eliminated by earlier rules; the solutions will be those combinations which remain after all the rules have been applied. Alternatively, a composite rule may be formed by linking all the rules by the connective *and*; this is an entirely equivalent procedure.

The Logical Machine of Jevons

In the paper of 1866, Jevons catalogued the complete set of possible combinations for various numbers of components up to six. This catalogue he referred to as the *Logical Alphabet*; he suggested that it might be permanently inscribed on a *Logical Slate* on which elimination problems could conveniently be worked. From this he derived the idea of a *Logical Abacus*, consisting of a number of separate wooden tablets arranged upon shelves of a frame; each tablet was inscribed with one of the possible combinations (e.g. *aBCD*), and pins were fixed at various points in the backs of the tablets in such a way as to facilitate the moving in one operation of all tablets bearing combinations which included any particular symbol (e.g. *a*).

It is clear that the idea of mechanizing the whole process was not far away; and in 1869 Jevons published two papers, the first of which (*The Substitution of Similars*) summarized his previous work, and the second (*The Mechanical Performance of Logical Inference*) disclosed that he had actually constructed a logical machine. This machine he demonstrated to the Royal Society on 20th January, 1870; it took the general form of his Logical Abacus plus a keyboard, from which by a most ingenious system of linkages, latches and rods the tablets were manipulated.

Only four components could be dealt with in this machine, and only a few of the basic logical relationships—those corresponding to "If *A*, then *B*" and similar forms with negatives, e.g. "If *a*, then *B*." Any number of rules of this nature could, however, be applied successively. The cycle of operations for the machine was as follows: all combinations were displayed in the initial state. If the first rule was "If *A*, then *b*," it was set up by depressing—

A on the left half of the keyboard
Copula in the centre
b on the right half.

Thereafter the depressing of a Full-stop key caused all combinations inconsistent with the rule (i.e. *ABCD*, *ABcD*, *ABCd*, *ABcd*) to disappear from sight by sliding upwards or downwards; at the same time the keyboard was restored to its original state ready for another rule. Subsequent rules eliminated further combinations; those remaining after all rules had been applied were solutions of the problem, i.e. consistent with all rules.

Although the restriction to a certain type of rule appears to impose a distinct limitation, this is not so great as might be imagined, since any of the sixteen possible relationships between two quantities may be expressed as the successive application of no more than two rules of the available form. For example—

$$A \text{ or else } B \text{ is equivalent to } \begin{cases} \text{If } A, \text{ then } b. \\ \text{If } B, \text{ then } a. \end{cases}$$

Venn Diagrams

A name worthy of mention is that of H. Venn, who in 1881 published a work, *Symbolic Logic*, which is still well known. Venn preferred diagrammatic to literal representations, and demonstrated the elimination principle by diagrams of characteristic form in which each combination of the components was represented by a cell of the diagram (Fig. 15/1). Elimination was carried out by shading the appropriate cells. Venn diagrams for more than four components are complicated and almost unmanageable. Venn stated explicitly that he did not hold logical machines in high regard; he did, however, go so far as to visualize making his diagrams in the form of a "jig-saw puzzle" to simplify their manipulation.

Modern Symbolic Logic

Great developments have taken place in symbolic logic since the early days. The symbolism of Boole has been shown to be a gross over-simplification; careful definition and close investigation of the philosophical aspects of logic have led to the development of a new synthetic language of great scope and considerable subtlety. Many brilliant workers have made contributions; the names of Frege, Peano, Whitehead, Russell, Wittgenstein and many others are well known. But these refinements have tended constantly away from "machine-like" principles; in consequence of this, even modern logical machines are still based on the simple Boolean concepts, which remain valid although limited in scope.

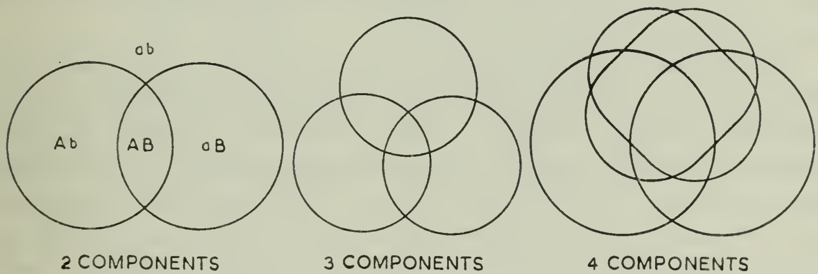


Fig. 15/1. Venn diagrams

Modern General-purpose Digital Computers

There is a close analogy between the idea of a two-state *logical* component and the *numerical* concept of a "binary digit"—a number which may only have one of the two values 0 or 1. Recent papers have shown how "information" may be measured by application of this concept.

Many modern general-purpose digital computers constructed mainly for the solution of *numerical* problems operate (for reasons of practical convenience) in binary notation. For this reason it is possible to use such machines to solve *logical* problems simply by taking advantage of the numerical-logical analogy. The method by which this was done is analysed in detail later in this chapter, where a machine* illustrating this principle is described.

These machines are, however, designed to solve numerical problems with great precision, and on this account tend to be large and complex. For logical problems much of this complexity is

* In 1949 a machine was developed by Ferranti, Ltd. at the Moston factory. This was described in a letter to *Nature*, part of which is reproduced at the end of this chapter.

superfluous, and small specialized machines designed specifically for logical problems have therefore been developed independently. Some of these machines are described in succeeding paragraphs.

Modern Specialized Logical Machines

The Kalin-Burkhart Logical Machine. In 1947 Kalin and Burkhart constructed at Harvard an electrical logical machine.* This machine was a direct electrical translation of the truth-table method; it was able to deal with up to twelve logical components and dealt successively with the 4,096 possible combinations of the states of these components. The rules were set up by means of multi-position switches; the available connectives were—*not*, *and*, *or*, *if then*, *if and only if*. The machine would select from the combinations all which were consistent with the rules set up.

The Ferranti Logical Computer. In 1950 there was constructed independently at the Edinburgh laboratory of Ferranti, Ltd. a seven-component logical machine. This was also a "serial" machine, dealing successively with the 128 combinations possible and selecting those consistent with the rules set up. The available connectives were: *not*, *and*, *or*, *or else*, *if then*, *if and only if*. Setting up of rules was by plug-and-socket connexions, the actual layout of the rule board being designed to give a pictorial view of the problem. The outputs of the rules went to a final *and* connective box which detected combinations consistent with all rules, indicating each by stopping the scanning mechanism to permit the solution to be noted. Various additional facilities were provided, including signal lights on the connective boxes to indicate when rules were individually satisfied, a manual switch for slow-motion operation, and so on.

This machine was able to solve problems of which the following is a simple example—

> I am going to University and have to decide what subjects I will take in my final year at school. The subjects I may choose are Mathematics, History, Science, English, Latin, German and French. English is compulsory. If I take Science, I must take Mathematics. If I take Latin I cannot take German as the time-table clashes. I do not want to take History. For entrance to the University I must have taken Science and French. What are my possible curricula?

* Allen Marquand suggested in 1885 an electrical analogue of Jevons' logical machine, and thereby anticipated Kalin and Burkhart by about sixty years. See W. Mays and D. P. Henry, *Nature*. Vol. 170, p. 696 (Oct., 1952).

The setting-up diagram for this problem is shown in Fig. 15/2; the solutions are as follows—

1. Mathematics, Science, English, French, Latin.
2. Mathematics, Science, English, French, German.
3. Mathematics, Science, English, French.

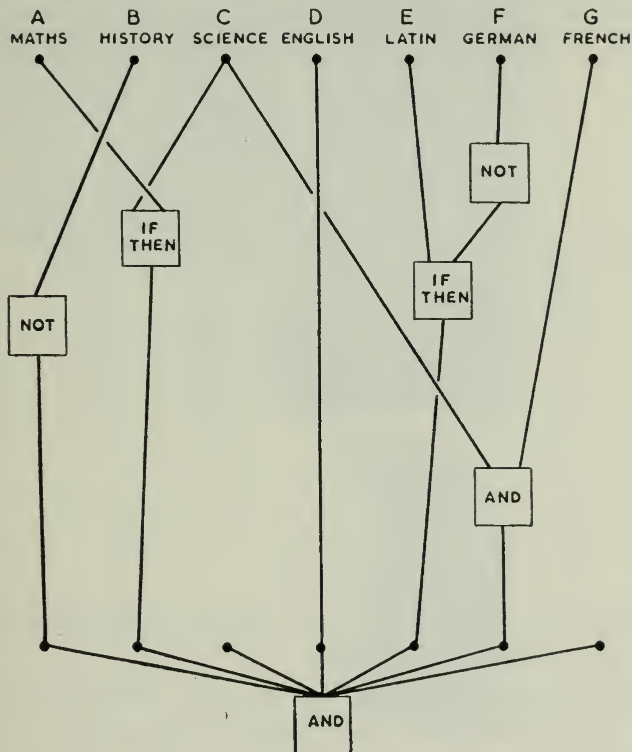


Fig. 15/2

Neither this nor the Kalin-Burkhart machine did appreciably more than Jevons' machine; the difference was merely that the development of electrical technique made it possible to perform the necessary operations in a tidier manner and with fewer moving parts. In both cases the number of connectives provided was sufficient to deal with problems couched in "ordinary" language without taxing overmuch the ingenuity of the operator. With a skilled operator, fewer would suffice; indeed, it was shown by Pierce in 1880 that one type (*neither A nor B*) is a primitive in terms of which all others may be expressed, by Sheffer in 1913 that another type (*a or b*) shares this property, and by Zylinsky in 1925 that there are no other primitives.

Many-component Problems

The investigation of logical problems containing a large number of components cannot be carried out in this way, since the number of combinations increases by a factor of two for each added component. Fifty components, for example, yield over one thousand million million combinations, which could not be scanned in any reasonable time even by modern high-speed techniques. Nevertheless, the human brain can readily conceive and even solve fifty-component problems of moderate complexity.

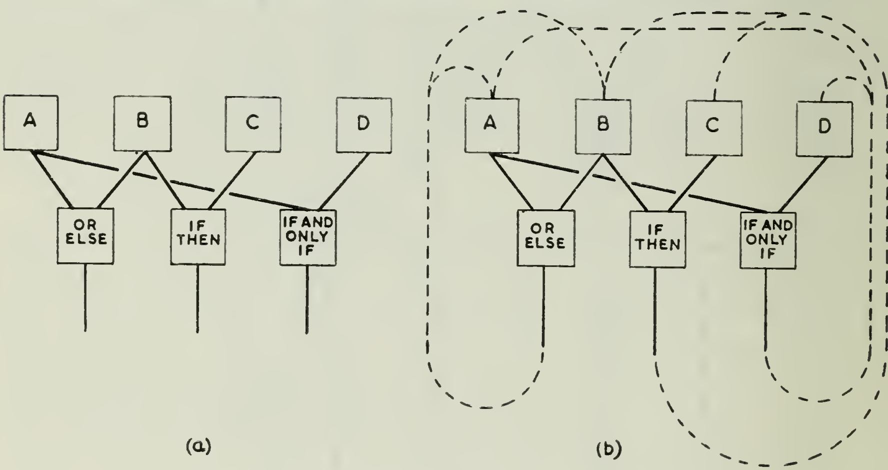


Fig. 15/3

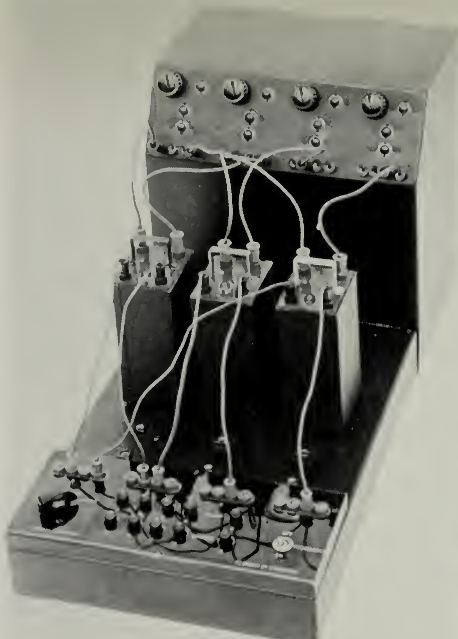
On this account attention has recently been directed towards methods of investigation of many-component problems; a method which has shown considerable promise is described in the next section.

The Feedback Principle

For the sake of example, let the problem be considered, the rules of which are—

If *B*, then *C*
A if and only if *D*
A or else *B*.

Suppose now that it is required to find not *all* combinations consistent with these rules, but only *one*. Let the problem be imagined set up on a machine of the type considered above, as in Fig. 15/3 (*a*), while some arbitrary combination is tested. At the output end are three leads, one for each rule; each of these leads



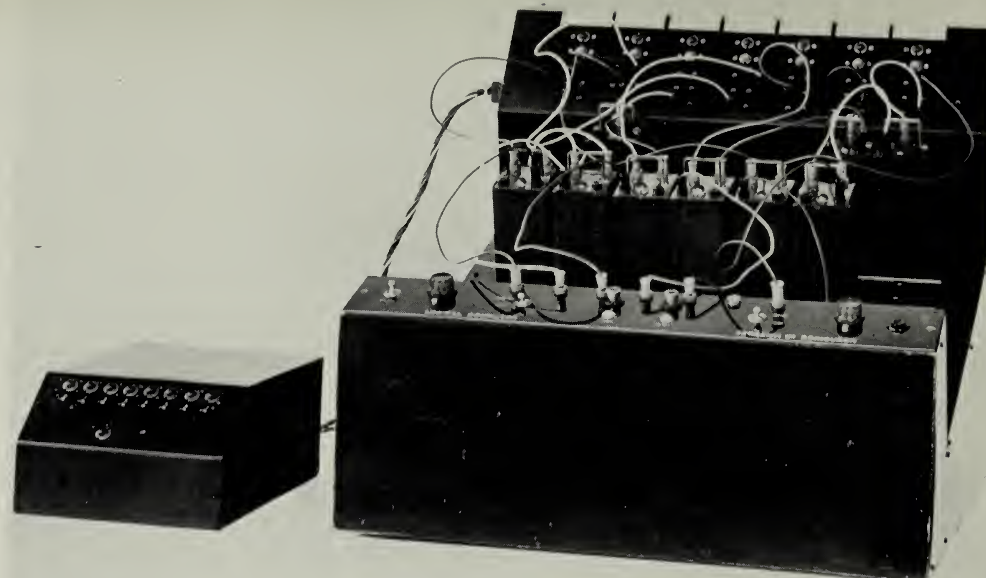


PLATE XVI. THE FERRANTI (EDINBURGH) COMPUTER

bears a signal which indicates whether its rule is "satisfied" by the combination being tested.

Now suppose these signals are "fed back" as in Fig. 15/3 (*b*), each to the components involved in the corresponding rule; and let it be arranged that the signal from any *unsatisfied* rule will alter the combination set up by changing the state of one of the components involved in that rule.

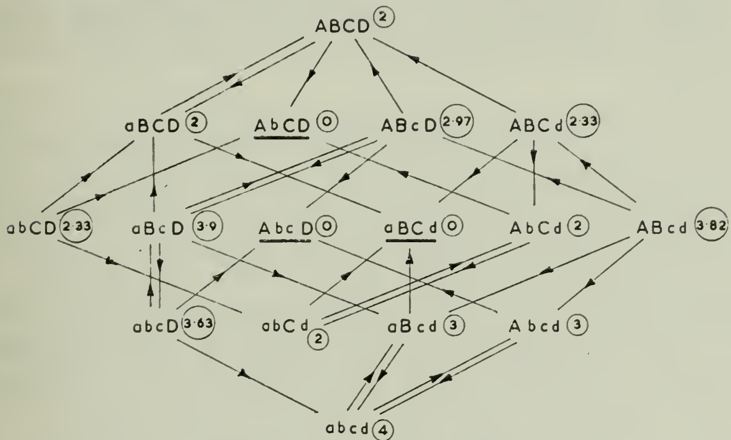


FIG. 15/4. A transition diagram

The effect of this is clear. If the combination set up is a solution of the problem, there will be no unsatisfied rules, no signals fed back, and no change. If it is not a solution some signal will be fed back and some change will occur. In other words, an electrical network has been set up, the stable states of which are identical with the solutions of the problem.

There is, of course, nothing to show that the network will reach a stable state in a reasonably short time, or indeed that it will ever reach one at all. As would be expected, the introduction of multiple feedback loops has raised a possibility of instability or continued oscillation. Such a computer might "argue in a circle" without ever reaching a solution.

Analysis of the Operation of a Feedback Computer. To study the behaviour of a feedback computer it is convenient to draw a "transition diagram" (Fig. 15/4). In this diagram all the possible combinations are set out in a convenient array and directed arrows drawn to indicate possible transitions. For example, the combination *ABCD* is inconsistent with the rule "*A* or else *B*," consistent with the others;

hence the fed back signals may demand change of *A* to *a*, or *B* to *b*; arrows are therefore drawn from *ABCD* to *aBCD* and to *AbCD*. Proceeding in this manner the diagram may be completed. No arrows lead *from* the solutions, which are underlined. The behaviour of the computer under any circumstances may be represented by a continuous path along the directed arrows.

It is clear that, in general, infinite paths along the directed arrows are possible. Whether they occur or not will depend upon the mechanism which decides upon the choice of path when alternatives are available.

There is a systematic mathematical technique for finding the outlet from a maze, which would certainly serve the purpose; this consists of taking the various paths from any point in a cyclic order on successive departures from that point; such a method will always trace a path on the transition diagram from any starting-point to some solution, provided that such a path exists. To make a mechanism which would provide for a computer actually to carry out this procedure in a many-component problem would, however, be clearly impossible, as all combinations passed through would require to be "remembered."

A less organized procedure, however, proves effective in the majority of cases, namely that of arranging for alternative paths from any point to be selected in a purely random manner governed only by determinate (in the simplest case, equal) probabilities. The behaviour of the computer then becomes that of a random process of the type known as a discrete Markoff process. Such processes have been extensively studied, and it is therefore possible to predict the average number of transitions in a computer programmed in this manner between any given starting-point and a solution. These values have been shown on Fig. 15/4 for the problem considered.

Calculations show that as the number of components increases, the relative advantage of the feedback computer over the "scanning" type previously described increases enormously. Still assuming that only *one* solution is sought, it may be shown that in a problem with n components and s solutions, a scanning computer will try on average $2^n/s$ combinations before finding a solution; the number of transitions in a feedback computer may, if the rules are of the most favourable kind, be as low as n/s. The difference between these quantities is startling; for example, if $n = 20$, $s = 2$, then—

$$2^n/s = 524{,}288 \qquad n/s = 10$$

so that the feedback computer may be 50,000 times faster.

The proviso "if the rules are of the most favourable kind" requires amplification. A feedback computer is sensitive to the degree of complication of the problem; simple situations with relatively "non-overlapping" rules, each involving few components, are dealt with quickly, whereas more "tangled" problems tend to take longer; this is somewhat akin to the behaviour of a human problem-solver. Fortunately the type of many-component problem met in practice is often not too involved in this respect, so that much of the potential advantage described above may actually be realized.

A Feedback Computer

Early in 1951 the first feedback computer was constructed at Ferranti Ltd., Edinburgh. It was a tiny four-component machine involving fewer than 20 relays in its construction; in addition to the "normal" type of operation described above, it could be operated in other modes so that different types of behaviour could be demonstrated; for example, it could be shown that if multiple transitions were possible (e.g. *A* and *B* both changing simultaneously if the rule "*A* or else *B*" were unsatisfied), instability would result. Push-button controls were provided by means of which the operator was enabled to set up any desired initial combination before connecting the feedbacks, which could all be connected simultaneously by throwing a single switch; buttons were also provided by means of which the operator could temporarily disconnect all feedbacks to any given component during the operation of the computer. An indication of the use of these controls is given below.

Plate XVI shows this computer with the problem described above set up on it.

Practical Use of Feedback Computers

It is clear that a feedback computer provides the best means of finding *one* solution of a many-component problem. For some applications this may be all that is required; in general, however, a complete set of all solutions will be needed. The question thus arises as to how to use a feedback computer to give the best approximation to this information in some reasonable time.

In any single trial, the initial condition (i.e. the combination first set up, which determines at which point of the transition diagram the random process begins) is at the operator's disposal. Generally speaking, the computer will tend to find a solution not greatly different from the initial condition. Hence successive trials from sufficiently different initial conditions will tend to give different solutions, if these exist. The idea of "sufficiently different" initial conditions is really a simple geometrical one: the 2^n combinations are the vertices of an n-dimensional cube in n-dimensional space in which each dimension represents one component, the two states of which correspond to two values (say 0 and 1) of the appropriate co-ordinate; the starting points required are the combinations representing a number of approximately equidistant vertices of this figure. The idea of "distance" in this connexion was noted by Cayley and Clifford in 1877. As an example, in a four-component problem the set of combinations—

abcd, *AbcD*, *ABCd*, *aBCD*

has the property required. In a many-component computer such sets could be permanently stored ready for use, and the operation of recording a solution and setting up a new initial combination could be made automatic.

The scope of the feedback computer may therefore be summarized thus—

(*a*) For problems of moderate complexity, it will find a single solution in a relatively short time.

(*b*) For problems whose solutions are few and isolated (in the geometrical sense described) steps may be taken to increase the chance of finding *all* solutions early in the operating cycle. This is particularly useful if the number of solutions happens to be known.

(*c*) For problems in which all answers must be found, all combinations must be set up; no advantage is therefore to be gained by using a computer of the feedback type.

When additional controls are provided by means of which the operator, by temporarily "clamping" certain components in one state, may cause the machine to explore only specified regions of the transition diagram, it is possible that a skilled operator may learn to assist the machine to avoid solutions already found, or to recover from partial instabilities. Such artifices might well be regarded as legitimate in practical application of machines of this type.

The Future of Logical Machines

Logical machines have up to now been regarded mainly as scientific curiosities. Practical applications have been few, and many of those which have been made have not been recognized as such because of their extreme simplicity—the ordinary double-switched electric light in the home, which possesses two switches A and B and lights when "A or else B" is depressed, is a logical machine. Nevertheless, scope for their application exists throughout our society, sometimes in the form of purely logical situations, and sometimes in logical-plus-numerical ones. Examples which come to mind are the checking for consistency of legal documents, rule books, political policy statements and the like; the checking of the "interlocking" of points and signals at railway junctions; the preparation of school time-tables to ensure that teacher and class and room are all available but not over-committed; the calculation of possible landing schedules at airports to avoid dangerous situations. Over and above this, there is likely to be a demand for logical machines of increasing complexity to replace the human operator in situations involving relatively low powers of judgment.

It may therefore confidently be predicted that the near future will see considerable advances in this recently re-discovered field.

THE USE OF GENERAL COMPUTERS FOR SOLVING LOGICAL PROBLEMS

We have already mentioned that logical problems can be solved on general-purpose computers of a type normally used for arithmetic computations; now we shall show in greater detail how this can be done.

Consider again the case of two statements (components), allowing four combinations or "classes," which we now write in the form of four columns—

k	0	1	2	3
	a	A	a	A
	b	b	B	B

counted by a "class index" or "column index" k.

Writing 0 for *false* and 1 for *true*, this table can be re-written in the following form—

k	0	1	2	3
A	0	1	0	1
B	0	0	1	1

and it may be noted that the content of column k, interpreted as a binary number with the least significant digit on top, is equal to the column index k.

The result of a logical operation, such as "A and B" (A & B) can now be expressed by adding another line, giving the table

k	0	1	2	3
A	0	1	0	1
B	0	0	1	1
A & B	0	0	0	1

since "A & B" is true (1) if both A and B are true (column 3), but false (0) in the other three cases.

Similarly the operation "A or B" ($A \vee B$) can be expressed by adding the line—

$A \vee B$ 0 1 1 1

the relation "A or else B," also called "A not-equivalent to B" ("$A \not\equiv B$") by—

$A \not\equiv B$ 0 1 1 0

the relation "A if and only if B," also called "A equivalent to B" ("$A \equiv B$") by—

$A \equiv B$ 1 0 0 1

and so on. In this manner, every result of the possible logical operations on A and B (including A and B themselves) is expressed by a row of four binary digits which can again be interpreted as a binary number. We may call it the *truth number* of the relation.

Logical operations can now be carried out by arithmetic operations on the digits of truth numbers.

Thus, negation is carried out by replacing every 0 by 1, and every 1 by 0. In particular, the truth number of a is 1 0 1 0 and of b 1 1 0 0.

The *and* operation corresponds to ordinary multiplication* of the digits, and for this reason, its result is often called *logical product* and written $A.B$.

The operation *or* is done by a process of modified addition, following the rules $0 + 0 = 0$, $1 + 0 = 0 + 1 = 1$, $1 + 1 = 1$. "A or B" is sometimes called the *logical sum* of A and B.

Non-equivalence corresponds to addition modulo 2, i.e. addition with the rules $0 + 0 = 0$, $1 + 0 = 0 + 1 = 1$, $1 + 1 = 0$.

* See also Chapter 2.

Equivalence is the negation of non-equivalence; its rules are $(0 \equiv 0) = (1 \equiv 1) = 1$, $(1 \equiv 0) = (0 \equiv 1) = 0$.

Similar rules apply to other operations. As an example, let us analyse the relation *not* $(a \& b)$.

From the truth numbers for	a	1 0 1 0
and	b	1 1 0 0
we find	$a \& b$	1 0 0 0
by digit multiplication, and the negation	*not* $(a \& b)$	0 1 1 1

by reversing the digits.

The result can be recognized as the truth number of "$A \vee B$", and this illustrates how the *or* operation can be synthesized from the *and* operation together with negation.

All these ideas can be generalized for the case of 3, 4, . . . n components. For 3 components A, B, C we have the *three-dimensional* truth numbers—

k	0	1	2	3	4	5	6	7
A	0	1	0	1	0	1	0	1
B	0	0	1	1	0	0	1	1
C	0	0	0	0	1	1	1	1

containing eight binary digits each; the complete table will consist of $2^8 = 256$ such numbers. In the n-dimensional case, we shall have 2^{2^n} truth numbers each containing 2^n digits.

Now consider a three-dimensional relation such as

R	0	1	0	0	1	0	0	1

It has a 1 in the places corresponding to $k = 1$, 4 and 7, or columns—

A	a	A
b	b	B
c	C	C

Using these columns, form the logical products—

k	0	1	2	3	4	5	6	7
$A.b.c$	0	1	0	0	0	0	0	0
$a.b.C$	0	0	0	0	1	0	0	0
$A.B.C$	0	0	0	0	0	0	0	1

and their logical sum—

$$A.b.c \vee a.b.C \vee A.B.C \quad 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1$$

using the rules given above.

It is then seen that the logical product of the components in column k has a 1 in digit place k, and a 0 everywhere else; and further, that the logical sum of these products is identical with the original relation R.

This is a particular example of a general rule. Every logical relation can be uniquely expanded in a logical sum of logical products (*Boolean expansion*), and the factors of these products are contained in the columns where the truth number of the given relation has a 1. Each of these products indicates a "solution" or "class" in the language used earlier on in this chapter.

In other words, to solve a logical problem it is only necessary to find the truth number of the logical relation expressing it. The solutions are then given by the 1-digits in this number. They can be found by looking at a table; but they can also be constructed from a knowledge of the column indices of the 1-digits.

We shall illustrate this method by the golf club example discussed in the beginning of this chapter.

The logical relation is $A \equiv (B \vee C)$.

Now we have—

k	0	1	2	3	4	5	6	7
B	0	0	1	1	0	0	1	1
C	0	0	0	0	1	1	1	1

Hence, by the rule for *or*—

$$B \vee C \quad 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1 = D \text{ (say)}$$

Further—

A	0	1	0	1	0	1	0	1
D	0	0	1	1	1	1	1	1

Hence, by the rule for equivalence—

$$A \equiv D \quad 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1$$

This number has a 1 in columns 0, 3, 5 and 7. The binary expressions of these numbers are (least significant digit on the left)

0	3	5	7
000	110	101	111

which gives, by writing small letters for o's and capital letters for 1's—

abC *ABc* *AbC* *ABC*

It is clear that arithmetic operations of this kind can be carried out on general-purpose computers provided their list of instructions contains facilities for acting on individual digits of binary numbers as required by the elementary logical operations. As a matter of fact, both the Manchester (Ferranti) and the T.R.E. machines are provided with facilities for carrying out the operations &, *or*, and $\not\equiv$. Most other machines contain at least the & operation since this has been found of general usefulness for carrying out ordinary mathematical calculations. On the other hand, the & operation, combined with negation, can be used to synthesize all other logical relations; negation can always be obtained by subtracting the given truth number from a suitable constant.

To test and demonstrate this system of dealing with logical problems, a small relay machine was built at the Moston Works of Ferranti Ltd. in 1949. This machine imitates the action of general purpose computers in so far as the result of an operation on two numbers can be stored, and the stored numbers can be used again as operands. A brief description of the machine has been given in a letter, dated 4 February, 1950, from W. Mays and D. G. Prinz, to *Nature*, an abstract of which is given here—

"The machine deals with the logical relations between three propositions or 'variables' *A*, *B*, *C*. These variables and their 'functions' (relations) are represented in the form of binary numbers (essentially an abbreviated form of their Boolean expansions) which are stored in electromagnetic relays, 'on' (1) standing for truth, 'off' (0) for falsehood. The state of each relay is indicated by a lamp.

"Five stores, each containing eight relays, are provided. Two of these (*U*, *V*) accept the initial propositions *A*, *B*, *C* and their negatives. A third store *S* accepts the result of a logical operation performed on the contents of *U* and *V*; these operations, which can be selected by corresponding switches, are conjunction (*and*), disjunction (*or*), implication (*if, then*), equivalence (*if and only if*) and negation (*not*).

"The contents of *S* can be transferred to two auxiliary stores and from there back to *U* and *V* so that further operations can be performed on them.

"A tautological relation is indicated by the lighting up of all the eight lamp of store *S*.

"A function $F(A, B, C)$ implies, in general, a number of functions $R(A, B)$ of two variables. It is then pertinent to ask for that particular function $G(A, B)$ among them which implies all the others. This 'optimum implicate' can be found by a process closely analogous to the projection of a three-dimensional pattern on a two-dimensional plane. Facilities for carrying out this process are provided in the machine."

Chapter 16

SPECIAL-PURPOSE AUTOMATIC COMPUTERS

Sir, We must beware of needless innovations, especially when guided by logic—Sir Winston Churchill

Very nearly all the automatic computers that have been designed in the past have been intended for scientific or similar use. The calculations to be performed have either been varied or very complex, and therefore the machines have been flexible in operation and can be defined as general-purpose computers. In practice the first electronic automatic computer, the E.N.I.A.C., was designed to solve ballistic problems and although it can be used to solve any problem it is much more efficient when employed in the field of ballistics. It might therefore be termed a special-purpose computer. It will be found that all machines are more efficient for solving some types of calculation than others but this fact does not make them special-purpose machines. The definition of a special-purpose computer is therefore rather vague and essentially arbitrary. The following definition, although not complete, probably covers most of the field.

"A special-purpose computer is a machine using automatic sequence control and digital techniques, in the design of which special measures have been taken to favour the efficiency of solution of a restricted range of problems at the expense of overall flexibility."

This definition also gives us some indication of the advantages to be gained by using special-purpose machines. From the design point of view, special circuits and techniques can be employed to improve speed of calculation and economy of design. These features in turn reduce size and cost and can probably lead to increased reliability. A further advantage is that if a restricted number of problems only can be solved, the operation of the machine is simplified. All the required programmes can be stored within the machine and can be selected by the use of a simple code number, by switches, or by remote control. Under these circumstances considerably less skill is required to operate the machine than to operate a normal desk calculator. One further advantage that is likely to be of use in commercial applications is that the machine can be designed to be completely self-checking, and made to check every calculation that is performed. This is likely to be of great importance in accountancy, costing, planning, etc., where mistakes are too expensive to be tolerated under any circumstances, and where considerable labour is employed with present methods of checking.

The Design of Special-purpose Computers

The complete special-purpose computer must be considered as consisting of three major sections—

1. The input and output system.
2. The storage system.
3. The computer itself.

Specialization of any one of these sections may cause the machine to be defined as a special-purpose computer.

The Input and Output System. This presents one of the major problems to be solved before special-purpose computers can achieve their full status. In most cases today information originates from a human agency and can therefore be generated only at human rates which are very slow. The obvious solution is that as much information as possible must be gathered automatically without human intervention, but this is only a partial solution, as for many years to come the machines must work in conjunction with human beings. A data-gathering system is therefore required which uses the minimum of human labour for transcription, etc., and in which the simplest possible actions are called for from the human originator of the information. We have many data-gathering systems available but all of them make too much use of humans to constitute a really efficient input system for an automatic computer.

Once the problem of input has been solved, the problem of output is not likely to be so difficult as machines can provide the output information rapidly and accurately in practically any required form.

Storage. Most present-day machines either do or could use three types of storage—

(*a*) A store to which access can be obtained rapidly and which is used during computation.

(*b*) A subsidiary store, larger than the fast-access store, in which information may be retained in bulk, and from which information can be transmitted in bulk to the fast-access store.

(*c*) The input and output medium which can be used on occasion to increase the storage capacity of the machine.

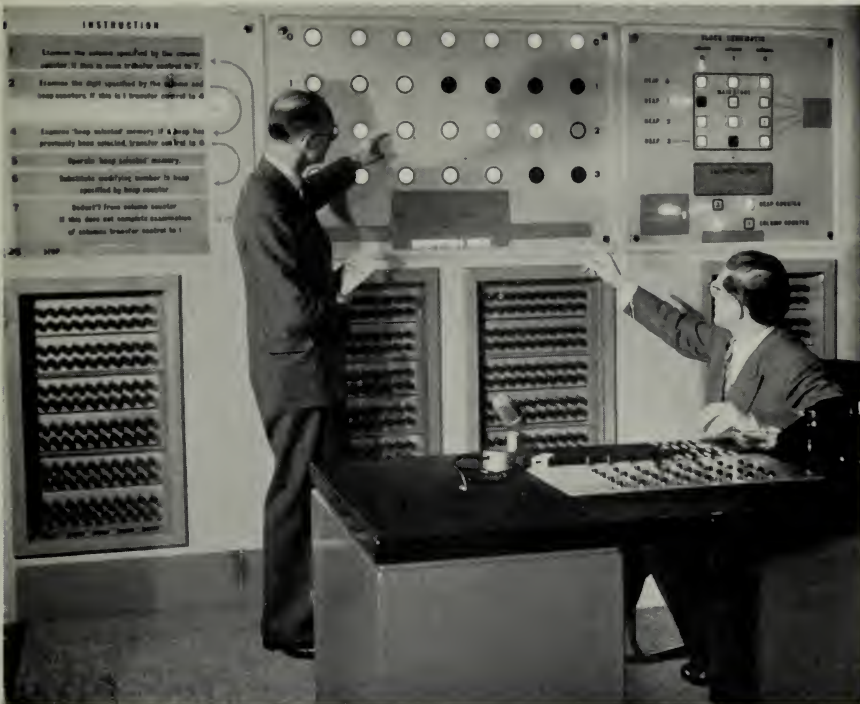


PLATE XVII. "NIMROD" AT THE SCIENCE EXHIBITION,
SOUTH KENSINGTON

Access to the fast store usually requires a few microseconds; to the subsidiary store, tens of milliseconds; to the input-output, the order of seconds; obviously an economic balance must be achieved. In a special-purpose machine this balance is likely to vary widely, and in place of the possible use of input-output as a store it may be necessary to supply a relatively long-access, large-capacity, library-type store for reference or storage purposes. It is quite possible that machines may be built in which this library is the most important section of the machine and the computer itself may be extremely simple.

The Computer. For many applications a general-purpose computer with special input-output and storage facilities may be quite sufficient. In addition to the special input-output a normal input-output might be provided so that the machine can be used as both a special- and a general-purpose device. If, however, a very restricted range of problems is to be solved, then a special-purpose computer section would also be needed. Under these circumstances considerable "trickery" can be employed. Mixed serial and parallel operation would have advantages for some applications, special arithmetic elements could be provided, and in general the design could diverge from that of a general-purpose machine considerably. Some idea of the possible techniques can be formed by considering the Nimrod computer,* shown in Plate XVII, which was designed for exhibition at the Festival of Britain, 1951.

Nimrod

This machine was designed to illustrate the principles of automatic computers in general, and did so by displaying an animated programme and block schematic diagram of the machine playing a game of *nim.* Since the displays which were used as output were fixed there was no necessity for making a general-purpose machine. For this reason the input mechanism was also fixed and a special-purpose computer was designed round the requirements of the programme. The programme that was used had been selected as being representative of a general-purpose machine playing nim but the final computer had no relation at all to a general machine. The storage requirements were small and therefore flip-flops† were used throughout for storage. Given flip-flop storage, counting operations could be performed in the actual storage elements. No common number channel was employed as the number of types of transfer

* See Chapter 23. † See Chapter 2.

was limited. In general binary numbers were used but it was found expedient in one section of the machine to use a quinary scale* and this was consequently employed. The machine was a parallel machine and not only transferred numbers in parallel but also performed several operations at once. Throughout the design of the machine in fact no consideration whatsoever was paid to normal computer organization. The total number of valves in the machine was 480, considerably less than would have been required for a general-purpose machine having the same number and storage capacity. A considerable speed gain was obtained since although the normal operating frequency was only 5 kc/s, the solution time was very much shorter than could have been obtained using a 100 kc/s serial, general-purpose machine.

In general Nimrod demonstrated quite well that if only one problem-type is to be solved then it may be possible to obtain large advantages by employing a special-purpose machine.

Possible Applications of Special-purpose Machines

There are obvious applications in industry and commerce for machines with limited flexibility if such machines are cheaper and easier to operate than a general-purpose computer. There would also seem to be some applications to control problems. In certain types of work rather complex or accurate servomechanisms could be employed with advantage. If the analogue computing elements in such servos were replaced by a digital computer, time sharing could be employed to reduce complexity, accuracy being relatively simple to achieve. Such applications may open up a new field for computing techniques.

It is possible that eventually digital operation may replace analogue operation in many applications. It is generally accepted that where high accuracy is desired digital techniques are always to be preferred. If the accuracy required is low then digital methods require more valves than analogue machines, but it must be remembered that whereas in an analogue device valve characteristics are quite important, in a digital device which has been correctly designed, valve parameters have very little effect on performance. Thus, in the interests of reliability, it may be profitable to use small special-purpose computers in place of analogue computing devices even though the change is accompanied by an increase in the total number of valves.

* A quinary scale is one based on the number 5, as a binary scale is based on 2, and a decimal on 10.

Chapter 17

DIGITAL COMPUTATION AND THE CRYSTALLOGRAPHER

The whole of chemistry, and with it crystallography, would become a branch of mathematical analysis which, like astronomy, taking its constants from observation, would enable us to predict the character of any new compound, and possibly the source from which its formation might be anticipated—CHARLES BABBAGE

THE BASIC TASK OF THE CRYSTALLOGRAPHER is to determine the atomic structure of molecules, organic and inorganic, from photographic patterns which result when regular arrangements of these molecules (i.e. crystals) are irradiated with X-rays. He may be given by the chemist several possible structures for a substance, and asked to choose from among them the one which will give the X-ray photographic pattern obtained experimentally—or, alternatively, from a single crystal of a substance, he may be asked to deduce the structure of the molecules composing it, with little advance information other than a very general idea of some of the atomic groupings.

The geometrical form of a crystal is the consequence of the regular arrangement of the molecules of which it is built up; the regularity is that of a three-dimensional pattern which is repeated over and over again in space. It has been recognized for many years that the geometry of such space patterns, together with the observed facts of crystal form and symmetry, could give much information about the internal structure of the molecules of which a crystal is built up; but it was not until the development of X-ray crystallography that it became possible to obtain information which could lead to the determination of the actual arrangement of the molecular units. Following the work of von Laue and the classic analyses of crystal structure by W. L. Bragg, this approach has now become standard, although unfortunately, as will be shown, it does not supply quite as much information as we would wish.

If a crystal is irradiated with a parallel beam of X-rays, the intensity of the diffracted radiation will vary in a manner which depends on the structure of the crystal, and the directions of the incident and diffracted radiation.

A photographic plate may be used to give a record of the intensities of this diffracted radiation; it is from this record that the crystallographer must unravel the details of the atomic arrangement of the crystal. The intensities of the diffracted radiation in certain selected directions are used to do this: these directions are specified by integral values of three parameters, h, k and l, which also depend on the crystal spacings and the direction of the incident radiation.(1)

Unfortunately, for a straightforward mathematical approach, a knowledge of the intensities alone is not sufficient to determine the information required. What would be desirable is a knowledge of both the amplitude of the X-rays and the relative phase for every value of h, k and l. The knowledge of phase, however, is not available and so the crystallographer must make do with what information he has.

It is at this point that experience in interpretation of other similar molecular structures enables him to assume certain arrangements of his molecules in the unit structure and, by calculation, to discover whether or not this arrangement gives rise to the pattern of intensities which he has actually observed. If his initial guess is sufficiently close to the correct result, then by a process which is so arranged that successive adjustments produce patterns of intensity closer and closer to the observed pattern, he can often arrive at the required atomic arrangement.

A calculation which is often of considerable assistance is the computation of what is known as a *Patterson synthesis*.(2) This synthesis can be calculated without any knowledge of the phase angles of the X-rays falling on the photographic plate, and gives the crystallographer some idea of the atomic distribution; it does not yield the final pattern which he requires, but may help him in deciding which configuration is a good starting point for the calculation mentioned above.

The sort of computation which is involved in this, and in many of the other processes employed by crystallographers, is generally a triple or double Fourier summation of the form—

$$\sum_{h=0}^{h_{\max}} \sum_{k=0}^{k_{\max}} \sum_{l=0}^{l_{\max}} \phi\,(h, k, l) \cos 2\pi \left(\frac{hx}{a} + \frac{ky}{b} + \frac{lz}{c}\right)$$

where h, k and l assume integral values only; the $\phi\,(h, k, l)$ are derived from the intensities measured for each value of h, k and l;

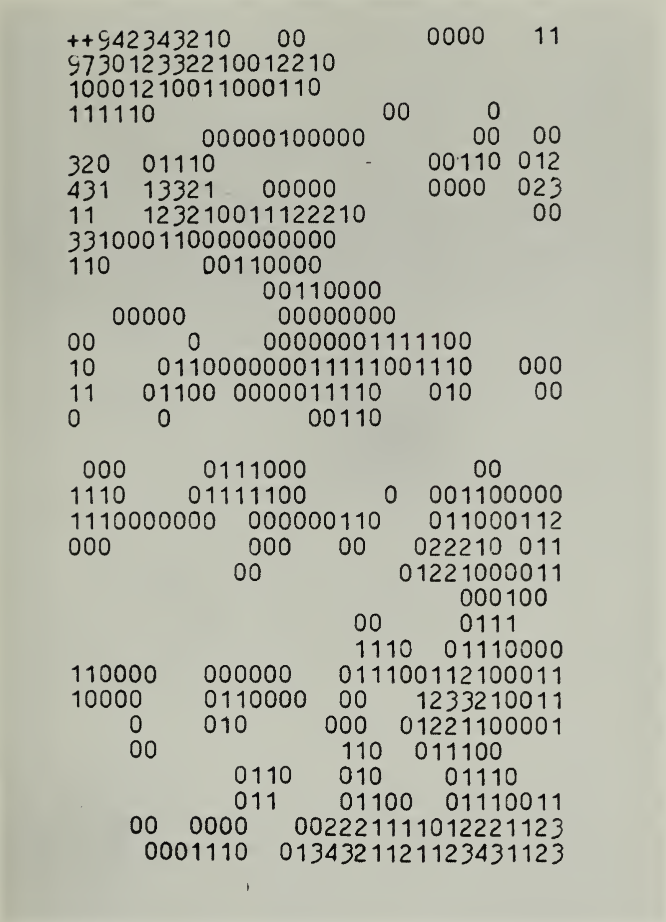


PLATE XVIII. A PHOTOGRAPH SHOWING THE *b* PATTERSON PROJECTION OF WHALE MYOGLOBIN PRINTED IN CONTOUR FORM

It has been arranged that—

(*a*) for every negative value of the function the machine prints a space

(*b*) for every positive value between 0 and 31 the number 0 is printed

(*c*) 1 is printed for values 32–63, 2 for 64–95, etc., up to 9 for values 288–319

(*d*) + is printed for values exceeding 319

Contours may rapidly be drawn at intervals of 32. Moreover, the programme is so arranged that, by changing two parameters, contours at any other interval $2m$ (where m is a positive integer) may be printed

and a, b and c are known constants, which depend on the crystal we are measuring.

In order to describe the type of programme required for dealing with a summation of this sort, we shall take it in its simplest form—the Patterson synthesis which occurs with some crystals possessing a high degree of symmetry. Under these circumstances, the summation reduces to—

$$\sum_{h=0}^{h_{max}} \sum_{k=0}^{k_{max}} \sum_{l=0}^{l_{max}} \phi\ (h, k, l) \cos 2\pi \frac{hx}{a} \cos 2\pi \frac{ky}{b} \cos 2\pi \frac{lz}{c}$$

and we require to evaluate this expression in a typical case for about 500 observed values of ϕ over something like 54,000 points, each point possessing separate (x, y, z) co-ordinates.

The values for which we evaluate our cosines are in this case evenly spaced and may be represented in a table of tabular intervals usually of about 3 degrees. This means that we need not expect our machine to compute cosines for each evaluation. We can compute a table at the beginning of the work, and hold it in the machine for the extraction of values as we require them.

The importance of operating from a table rather than the alternative of computing each value separately may be gauged from the time taken in a typical calculation, using both methods. The details of the size of the calculation or the particular machine on which it was done need not concern us here—the ratios of the times alone are important. If we compute the cosines separately as we progress by means of a formula such as—

$$\cos (A + B) = \cos A \cos B - \sin A \sin B$$

the time taken per value printed out is in the vicinity of fifteen seconds. If we form a table initially, the time taken per value is twelve seconds—assuming that a quadrant is divided into thirty tabular intervals of three degrees.

Because of the added convenience derived from operating with numbers which fit into the scale of two, it becomes easier to work with the table of thirty-two (2^5) tabular intervals rather than thirty. If this is done, the same problem will be carried out at a rate of 10·5 sec per point.

The easiest way of describing the form in which the Fourier summations are carried out (i.e. the "flow sheet") involves consideration of the two-dimensional synthesis—a very useful computation which corresponds to the three-dimensional case for $z = 0$. In two dimensions the Fourier summation becomes—

$$\sum_{h=0}^{h_{max}} \sum_{k=0}^{k_{max}} \phi(h, k) \cos 2\pi \left(\frac{hx}{a} + \frac{ky}{b}\right)$$

And if we form, for each value of x and y, the quantity—

$$2\pi \left(\frac{hx}{a} + \frac{ky}{b}\right)$$

for each value of h and k in turn, then extract its cosine, and multiply the resulting value by $\phi\ (h, k)$, we can find the contribution of each

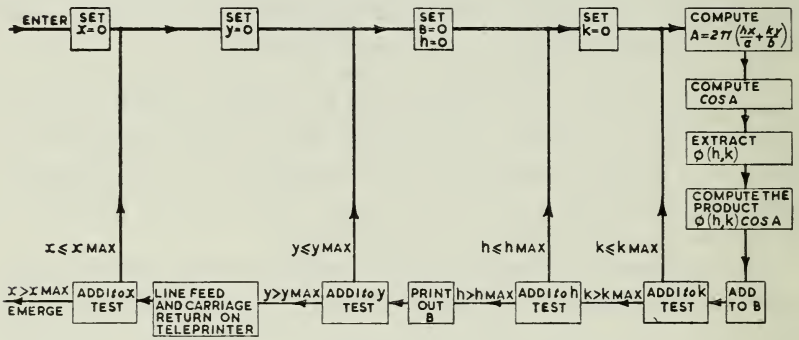


Fig. 17/1

$\phi\ (h, k)$ to the total sum. In this case, the flow sheet will be that shown in Fig. 17/1.

The figure is self-explanatory; it should be pointed out that the "print" subroutine is included between the h and y loops because the counters will indicate at this point that the double summation is complete.

This method is not the most efficient way of doing the work, however; an increase in speed of approximately $1/h_{max}$, where h_{max} is the maximum value of h for which F_{hk} values are available, is possible if we compute in turn the k quantities—

$$A(k, x) = \sum_{h=0}^{h_{max}} \phi\ (h, k) \cos 2\pi \frac{hx}{a}$$

and then perform the final summation—

$$\sum_{k=0}^{k_{max}} A(k, x) \cos 2\pi \frac{ky}{b}$$

If the $(k_{max} + 1)$ terms $A(k, x)$ are stored for each value of k, the programme may be arranged to print out in turn all the y values corresponding to any one value of x. The equivalent flow sheet is shown in Fig. 17/2.

When computed in this way, the example for which times were quoted above was completed at a rate of 2·5 sec per point, of which about 1·3 sec was printing time.

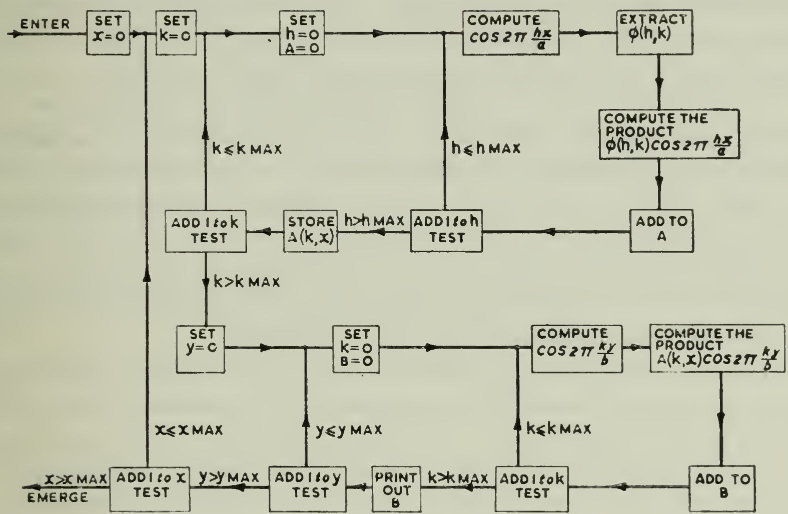


Fig. 17/2

This question of printing time assumes considerable importance in calculations of this type. A typical estimate indicates that, for a three-dimensional summation involving about 500 input values and 54,000 output values, the computation time expected for the whole problem will be of the order of four and a half hours for a computer whose basic operation time is one millisecond; the time required to print out relevant information will be about nine hours if single teleprinter output is used. Problems of this type approximate more closely to commercial calculations than to large-scale scientific calculations, as the ratio of input-output time to computing time is sufficiently large to be of major importance; in consequence we are driven either to economize on the printed information which we expect the machine to yield, or to utilize faster output devices.

Thus, in the example mentioned above, if we were to use a seventy-character parallel-printing output of the type used in Hollerith machines we would expect the printing time to be reduced to about three-quarters of an hour; this printing can moreover take place while computation is going on.

With regard to economizing on the amount of information which we expect our machine to yield, we can take advantage of the fact that the numerical results obtained are used very frequently for plotting contour maps of, say, electron density. Plate XVIII gives an example of how, with a little ingenuity, we can virtually make our machine do its own contouring. This is arranged by making our programme identify the results of a computation as lying between two contour levels, and print out an appropriate symbol.

The final determination of crystal structure in any but the simplest cases is the result of a lengthy—if systematic—trial-and-error process, of which computations of the type mentioned above form only the "central loop." We have shown how these computations may be performed as accurately as we require; the times taken are considerably faster than the corresponding times for hand computation or even computation with punch-card machines, and should be of considerable assistance for many purposes.

A recent computation similar to the type described above was carried out on the Manchester computer by members of the staff of the Chemistry Department of Leeds University. This department has had considerable experience with punch-card equipment for the same class of computation; since this was their first experiment with a digital computer they did everything twice as a check. Nevertheless the computer was more than forty times as fast as their punch-card installation. Had certain approximations been made, the ratio might have been reduced to about thirty to one; the saving with punch-card equipment would have been higher than with the computer. The Leeds University punch-card installation consisted of a Hollerith Senior Total Tabulator, a Sorter and a Reproducer, and has been found to be between six and ten times as fast as hand computation for crystallographic work.

Punching time for cards and for teleprinter tape is about the same, and represents about four times the period required for the complete computation with the computer.

In the 1930's it was quite normal for a research worker to spend three months in measuring an X-ray plate, and three years in analysing his results. Modern computers, as we have shown, allow the analysis to keep pace with measurement, and, moreover, make it possible for the first time to undertake the analysis of quite complicated molecules.

However, the total times taken in useful cases are such that, at least until digital computers are speeded up by quite a large factor, the crystallographer must still use a considerable amount of personal judgment and experience in achieving his final object; automatic computers are for him one more useful tool which will free him from the drudgery of routine calculations.

REFERENCES

1. Booth, A. D. *Fourier Technique in X-ray Organic Structure Analysis.* Cambridge University Press (1948).
2. Bennett, J. M. and Kendrew, J. C. "The computation of Fourier syntheses with a digital electronic calculating machine." *Acta Crystallographica* **5** (1952) 115–6.

Chapter 18

THE USE OF HIGH-SPEED COMPUTING MACHINES IN METEOROLOGY

I accept the universe—Margaret Fuller (of Boston)
Gad! she'd better—Carlyle

To most people meteorology means the science of weather forecasting. This is perhaps unfortunate because while meteorology is a science, weather forecasting remains to some extent an art. The scientist is not normally expected to say anything about a question if he does not know the answer, but in meteorology, as in medicine, the scientist is always called upon to give an answer of some sort. The public is not content to be told "an answer to this question cannot be given in the present state of knowledge." If one doctor can prescribe no treatment the patient will go to another who will, and rather than receive no advice he prefers some opinion even if it is no more than a rash guess, for he hopes it will be an enlightened guess. Forecasts are now made as a routine but the meteorologist is continually conscious of the limitations of his knowledge. Thus, more time is occupied in giving opinions about problems which cannot at present be solved with certainty than in improving our basic knowledge.

The pressing need is to improve our fundamental understanding of the atmosphere so that forecasting will involve more exact science and less guesswork. Before turning to this aspect, we may ask: Could machines be used on the basis of present knowledge to produce forecasts which have not been attempted hitherto because of the enormous amount of labour involved? The time available for making a forecast is strictly limited; is it this that has made it necessary to turn forecasting into an art, and can we hope for more scientific forecasts now that electronic machines can carry out all the necessary calculations in a few minutes?

It was thought, thirty years ago, that all the basic physical laws which the atmosphere obeys were already known or soon would be, so that to prepare a forecast one would merely have to solve a system of equations. The numbers going into the equations would describe the weather up to the present, the equations themselves would describe what processes the atmosphere undergoes, and the solution that would come out would describe the future weather. It was, of course, realized that this would be an extremely laborious process, and so no one seriously considered it except as a method to be used when and if a computing engine of great power became available. In the years that have followed, this optimism, based on nineteenth century mechanistic ideas, has been replaced by a much more realistic appreciation of the problem.

The present century has thrown up many new concepts which have enormously simplified our thinking about the atmosphere. There is no space to discuss these, but we have also found, as more and more observations have been accumulated, that the atmosphere is lumpy; the wind is gusty, the temperature is continually fluctuating; cloud masses show how irregularly moisture is distributed. Such facts were vaguely realized but it was thought that they would average out to something quite comprehensible and their effect could be fairly simply represented in the equations. They were mere fluctuations around the mean values, the mean values being the important things which would be described by the equations. But before the equations can be used the mean values must be precisely defined. If the earth wore the mean motion, called *the general circulation*, as a recognizable garment, with a sort of embroidery of turbulence or lumpiness, then all would be well; but as more and more observations became available the embroidery grew until now it seems that the weather over most of the earth consists mainly of embroidery and the main garment of the general circulation, if it exists, is completely obscured over vast areas, of which the North Atlantic ocean and Europe form one.

Thus, although the same physical and dynamical laws apply to the embroidery as to the main garment, the application of them becomes impossibly complicated. Not only does the prospect of being able to formulate the equations for a hypothetical machine seem ever more distant, but it seems increasingly unlikely that enough observations will be available on which to base the calculations. How fine is to be the net of reporting stations when we know that not only from day to day and hour to hour, but also from minute to minute and indeed from second to second the wind, the temperature, the humidity and the pressure fluctuate unceasingly? If we send up a balloon, how much would the measurements it makes have differed if we had sent it up a few miles away or a few minutes later? We simply cannot obtain observations that are completely representative of the surrounding air, so that even on the broad scale a calculated forecast would soon go wrong. One and a half days is usually regarded as the limit beyond which we cannot expect to forecast in any detail in these latitudes, and often the forecaster cannot predict as far ahead as that.

The question that arises is therefore whether forecasts within this period can be improved if a machine is available to do any computation desired. It is too early to dogmatize, but it seems most likely that to use a machine we must choose occasions on which only a few of the many influences that are at work in the atmosphere are operating, preferably the ones that are best understood, for then we have some hope of formulating the problem. So far this has only been attempted for one kind of situation and its success cannot yet be gauged. The relation between pressure and wind is described in well known equations, and so if we can find an occasion when friction, cloud, the heating and cooling of day and night, and large mountain ranges have only a small or distant effect, and when the embroidery mentioned above is not too complicated, then a forecast could be made mechanically. The most obvious example of this is the motion of the air at high levels across the Atlantic. This possesses economic as well as theoretical interest, for air travel in the stratosphere offers many attractions, not least being the absence of cloud and of changes in the weather. The objective is simply to know the wind in advance for navigation purposes.

Forecasters are mainly concerned to estimate as accurately as possible the distribution of atmospheric pressure 24 hours after the most recent complete chart, i.e. about 20 hours after the time the estimate is made. If the situation is a simple one it is just possible that present methods could be improved upon by attempting the whole problem on a machine. This supposes that equations can be set up to describe the atmospheric processes and that all the information necessary to solve them can be expressed in numerical form. There are two schools of thought about the best approach to be made, though it is the effort rather than the opinion that is divided; each school is following the work of the other almost as much as its own. On the one hand we could solve the problem completely in a simple case, and having learned to do that, make it more complicated, at each advance getting nearer and nearer to the real problem in the atmosphere, in the belief that ultimately the forecast chart will be produced by a machine every day, instead of only every now and then when the situation is a simple one. On the other hand we can formulate bits of the problem and get the machine to solve them, filling in the rest by experience and all the techniques at present used. Thus for instance a formula has been derived which describes the amount of deepening of depressions that will take place. The computation of the formula is laborious and is not used by forecasters because the calculations cannot be made in time to be useful. Also, since they are laborious, it will be a long time before the formula has been tested extensively to see if it would have been helpful. A high-speed computer could therefore be used in the first place to test it in past cases and then, if it showed promise, to place the results of the calculation before the forecaster each time he makes a forecast. This would place one more tool in his hands, which he could use or not at his discretion when he has learned by experience in what kind of situation the formula is really helpful. This second school of thought, therefore, never proposes to do the forecast mechanically but merely to place before the forecaster the results of calculations for him to use in creating his forecast. The art in forecasting will still remain. Every machine, however complicated, can be operated with more or less skill, and the same can be said of the results given by calculating formulae which are known to be approximate only; the skilled forecaster will know what reliance to place upon the predictions made by the machine. When one formula has come into regular use others will follow, and perhaps before long a machine will be employed regularly at certain times of day on computations for forecasters.

We now turn to the question of improving our understanding of the processes of the weather. Our objective is to find some way of describing or representing the processes more simply than we can at present. For instance, we have a set of equations which tells us how much water is condensed out as cloud droplets when air is lifted to higher levels. In theory we could solve these equations every time we wanted to study a special case, but this would be very laborious and would take so much time that the forecaster would never do it. Instead solutions of the equations have been obtained for several cases and are represented on a diagram; the diagram is then used by the forecaster, and from it he can obtain all the information he requires in a few minutes. He can calculate the height of thunder-clouds, heights at which the barometric pressure takes certain values (which is useful in calculating winds), the probable maximum day temperature, and a variety of other useful things from this diagram. The diagram, as at present calculated, is accurate enough for normal purposes but it is of considerable interest to compute it again with much greater accuracy and with some extra refinements, so this work is being carried out on an electronic computer because of the enormous amount of labour that would otherwise be required.

Several other such diagrams are contemplated on which it is hoped to represent various other physical properties of the atmosphere, so that when the observations are displayed on them all the necessary deductions can be made quickly. Large quantities of heat enter the atmosphere as sunlight and are lost in space as infra-red radiation, but the amount entering or leaving depends upon the distribution of the various constituents of the atmosphere, notably water vapour and ozone. With high-speed computers available, it will be possible to undertake a programme of calculating the radiative properties of the air for a great variety of distributions of these constituents, and then to deduce the properties of the atmosphere of the moment from the calculated cases. The results might take diagrammatic form or might be available as a large book of tables like the *Nautical Almanac* or tables of logarithms. Without a machine to do the computing, this task will certainly not be undertaken.

Another way in which machines can come to our aid is to enable us to deduce more from our observations. For instance, it is possible to make certain spectroscopic measurements from which, in theory, we can deduce how the ozone is distributed overhead. If we could complete this task a powerful new method of measuring air movements at very great heights would be available because it is at those heights that the ozone mainly exists. The actual observations are made on the ground; these are very simple, and measure the way in which sunlight is modified as it passes through the ozone, but to deduce the distribution of ozone is not easy. It could be done with a high-speed computer and a programme to do this is under consideration.

This example raises the question of whether a high-speed computer would be mainly useful to carry out certain research projects once and for all or to make daily computations so that the machine would be in the whole-time service of forecasters. It is too early for forecasters to say that a machine ought to be built for them now, but there seems little doubt that when the research projects have yielded their fruit our knowledge of the basic processes will have been sufficiently simplified for many of the calculations a forecaster will then wish to carry out to be mechanized; but even then forecasting will remain to some extent an art.

Every year some new phenomenon comes to light. The rosy dream of smooth jet-propelled air travel high above all storms has been somewhat darkened by the discovery that even the clear air of the stratosphere can be uncomfortably bumpy at times. This was unexpected, and before we can forecast these bumpy regions we must explain the phenomenon. Various theories must be tested, and to test some we must do a lot of computing; here again the machines can help us.

From time to time the question is asked "Can we control the weather?" We need only to calculate the energy of the average storm to realize that we have little hope of creating the weather on anything but a minute scale.* We can stir the air by fans to protect fruit crops from frost, we can disperse fogs by F.I.D.O., but these are only local phenomena. All we can hope to do is by some triggering action to set a process going which will change the weather. The example much in the public mind at present is the problem of making the clouds which often occur over dry regions of the earth yield their moisture as rain. We must make the many tiny cloud droplets that remain suspended in the air coagulate into a few big ones which will fall out. With a view to understanding the processes which cause this to happen in places where it does rain, calculations have been and are continuing to be undertaken to discover what will happen if certain substances are introduced into the clouds. Experiments alone cannot give the answer because in cases of success we cannot be certain that it would not have rained anyway! Here again an electronic high-speed computer is being used to solve the differential equations which describe the growth of a big raindrop as it sweeps up smaller ones by falling faster than they do.

In conclusion we may say that machines will enable us to understand the atmospheric processes better and produce more reliable forecasts for the next 24 hours, but beyond two or three days we cannot at present see how they can help. Longer term questions like "Will this be a wet, dry, cold, or sunny season?" have not been put on a quantitative basis, which is a first requirement before a machine is used. Neither can we guess sufficiently well the reasons why some seasons are wetter or hotter than usual for us to know what calculations we ought to make.

* The *organized* kinetic energy of an ordinary depression in the latitude of Great Britain is equal to the energy of about ten thousand atomic bombs.

Chapter 19

AN APPLICATION TO BALLISTICS

Zeno argued that, since an arrow at each moment simply is where it is, therefore the arrow in its flight is always at rest. Bergson meets this argument by denying that the arrow is ever anywhere—BERTRAND RUSSELL

THIS CHAPTER WILL BE DEVOTED to a description of the way in which electronic digital computers may be used in the solution of the problem of external ballistics. External ballistics is that branch of applied physics dealing with the motion of projectiles outside the projecting mechanism, and is one of the oldest studies in applied science, having been studied by scientists since at least the time of Galileo. The studies of the pre-Newtonian scientists in this field provided many striking examples of the phenomena of the motion of bodies, later to be accounted for in Newton's Laws of Motion. Today, the study of external ballistics is devoted mainly to the application of the knowledge of the physical facts, which are well understood, to the production of firing tables for use in gunnery.

The difficulties that arise in this application are caused by the fact that the methods of exact mathematical analysis may not be used, since the problem cannot be formulated in detail without recourse to a numerical description of the physical facts. That is to say, much of the description of the phenomena is given not as exact formulae, like formulae in algebra, but in the form of tables of numbers. An example of the description of events in the form of a table is furnished by those railway timetables that give the mileage of the stations from London as well as the time of the train; with the aid of one of these one could calculate the average speed of the train over the sections between the stations, but not the speed at any particular point, as would be possible if the distance travelled by the train were expressed exactly, as, for instance, by an algebraic formula. Problems such as the external ballistics problem that are defined, at least in part, by tabulated information, may be solved only by the use of the methods of numerical analysis, methods which require much computation. It is now possible to use electronic digital computers to perform such work, which in the past has been performed by human computers with great labour and loss of time.

It was primarily to perform such computations that the first electronic digital computer was constructed. This was the E.N.I.A.C. It has been used for a variety of problems, including, among more mundane uses, the calculation of the constants π and e to two thousand places of decimals. The nature of this computer, although its capabilities are small compared with some of the more recently constructed machines, enables it to be used in the solution of a large class of problems capable of mathematical description in terms of ordinary differential equations, of which the equations of external ballistics may be taken as an example. Although the urgent need for solutions to the problem of external ballistics had passed by the time this machine was put into service, the problem may be taken as an example, since the phenomena met in this field are well known and require no special knowledge for their comprehension. Many problems mathematically similar to the problem of external ballistics occur in dynamics, astronomy, engineering and many other subjects, and their solution by means of computing machines is similar.

In gunnery it is necessary to know the angle of elevation of the gun for the range required. This is obtained from a firing table for the gun, which shows the angle of elevation required for various ranges under standard conditions of air temperature, wind speed and various other factors affecting the range, and also lists the corrections required for non-standard conditions. It enables all factors affecting the range to be taken into account; these include the muzzle velocity of the gun, the atmospheric conditions of wind and air temperature (which may vary in unusual ways through the various layers of the atmosphere traversed by the shell), as well as the heights of the gun and the target. Although it is usually required to know only the elevation, it is sometimes necessary to know other details of the trajectory, or path of the shell, such as the time taken by the shell to reach the target, or the velocity or direction of impact. It will be seen that the amount of information required is large, too large to be obtained solely by experimental firings: it is in fact obtained by computing mathematically the trajectories for a set of values of angle of elevation, muzzle velocity, etc., relevant to the gun for which the firing table is required. Information as to the muzzle velocity of the gun and the air resistance of the shell used in the computations is obtained experimentally from a relatively small number of firings and forms the basis of the whole firing table which is built up by mathematical methods from the results.

For the purposes of the calculation of the trajectory a shell is assumed to behave as a body in motion under the influence of two forces only, namely the force of gravity, acting in the vertical direction, and the force of the air resistance opposing the motion of the shell, acting in the direction opposite to motion of the shell at all points. The force of gravity is constant but the air resistance depends in a complicated manner on the velocity of the shell, being small at velocities much below the speed of sound, rising extremely rapidly as the speed of sound is approached, but thereafter not changing so quickly. The mathematical equations to be solved are those found by applying Newton's second law of motion to the horizontal and vertical motions of the shell separately; these equations, which must be solved simultaneously, relate the acceleration, the mass, and the force acting on the shell. The forces acting vertically are firstly, gravity, and secondly, a component of the air-resistance force; the force acting horizontally is the horizontal component of the air-resistance force. These forces are equated to the product of the mass and the acceleration (vertical or horizontal), resulting in two equations—

$$m \times \text{vertical acceleration} = -mg - (\text{vertical component of } F)$$

$$m \times \text{horizontal acceleration} = -(\text{horizontal component of } F)$$

where m is the mass of the shell, g the gravitational acceleration and F the air-resistance force, which acts in a direction opposite to that of the motion of the shell at any point. The negative signs on the right-hand side of the equations express the fact that the forces retard the motion of the shell. There is no equation representing the motion of the shell at right angles to the plane of flight, since it has been assumed that there are no deflecting forces.

If the equations are divided by the mass m they become—

$$\text{Vertical acceleration} = -g - (\text{vertical component of } F)/m$$

$$\text{Horizontal acceleration} = -(\text{horizontal component of } F)/m$$

The vertical component of F is equal to F times the sine of the angle between the instantaneous direction of the shell and the horizontal, and the horizontal component is equal to F times the cosine of this angle. Similarly the vertical and horizontal components of the velocity are equal to the velocity multiplied respectively by the sine and cosine of this angle. We use this fact to express the components of F in terms of F itself and the components of velocity; since the components of F are to F as the components of V are to V, the equations become

Vertical acceleration $= -g -$ (vertical velocity) $\times F/(mV)$

Horizontal acceleration $= -$ (horizontal velocity) $\times F/(mV)$

where V is the true velocity of the shell. Experience has shown that further simplification may be made by expressing the quantity $F/(mV)$ as the product of three terms, one depending on the velocity of the shell, another on the density of the air and the third on the shape and mass of the shell.

It would serve no useful purpose to give here the exact form of the simplified equations. In the final form they relate the following quantities—

(*a*) The vertical and horizontal components of acceleration.
(*b*) The vertical and horizontal components of velocity.
(*c*) The density of the air.
(*d*) The function representing the air resistance.
(*e*) The gravitational acceleration.
(*f*) The ballistic coefficient.
(*g*) The vertical and horizontal co-ordinates, giving the position of the shell.

Certain of these quantities, for instance the ballistic coefficient and the air-resistance function, require explanation. They arise in the specification of the air-resistance force in terms of the ballistic coefficient and the air-resistance function, representing the dependence of the air resistance respectively on the properties such as the mass and the shape of the shell and on the velocity. For a wide variety of shells the resistance function is dependent only on the velocity and on no other factor. It is due to the definition of this function by tables of its value for a set of values of the velocity that it is necessary to use numerical methods. This function, which is obtained empirically, is usually tabulated for convenience as a function of the square of the velocity.

The density of the air depends only on the height, or vertical co-ordinate of the shell, and is usually taken to be an exponential function of the height; that is, the rate of decrease of density with height is proportional to the density. The ballistic coefficient is a constant throughout the calculation.

From this point onwards, the problem is that of solving (or integrating) the differential equations mentioned above. Fortunately the method employed for this is quite simple and, at least in essentials, may be described in non-mathematical terms. This is because the terms used are familiar concepts such as acceleration and velocity. The method is based on the simple fact that, whereas it is not possible to write down an exact solution of the problem as a whole, it is possible to achieve a solution step by step; while it is impossible to work out directly where the shell will fall for given conditions of firing, it is possible to forecast the flight of the shell over small portions of its trajectory with an accuracy that increases as the portion is made smaller. The flight of the shell is forecast from point to point over the whole trajectory and if sufficient points are taken along the path, the forecast of the point of impact of the shell on the target may be made with the desired accuracy. Thus if the shell takes five seconds to complete its flight, the trajectory might be divided up into portions traversed by the shell in one fortieth of a second, and forecasts made from one point to another, the forecast for one point using the results of the forecast for the previous point till the two-hundredth point is reached. The first forecast is made from knowledge of the vertical and horizontal velocities of the shell as it leaves the muzzle of the gun, and subsequent forecasts are made by exactly the same method but using the vertical and horizontal velocities of the shell at the beginning of the portion of the trajectory considered, instead of the muzzle velocities. The two hundred steps of the trajectory require the same calculation but in each case the numbers used in the calculation are different. The process of calculation is a repetitive one with, in this example, two hundred repetitions. In practice the number of steps is never known at the beginning; the solution is carried through with certain sizes of step until it is completed. The end of the process is indicated when the vertical co-ordinate (the height of the shell) falls to zero relative to the height of the gun, or of the target if the gun and target are not at the same height. A repetitive process such as this is ideally suited for automatic calculation since, once the machine has been set up to do one step of the calculation, additional instructions that cause the machine to repeat the step will enable the whole calculation to be made automatically. The equations to be solved are differential equations (that is, they express relations between differential coefficients) since the acceleration is the differential of the velocity against time, and the velocity is the differential of the position of the shell against time. However, automatic digital computers have as little aptitude as the average person for differential calculus; in fact the only way it is possible to use such a computer to solve differential equations is to use arithmetic approximations to the differential coefficients that occur. There is a great variety of such approximations which might be employed for solving the equations of motion over the small intervals described.

One method, similar to that used with the E.N.I.A.C., is based on the assumption that the forces acting on the shell over the interval are constant. The estimation of the error introduced by this assumption requires an application of the differential calculus, but a description of the method itself is purely arithmetical and quite simple.

Suppose that during the interval under consideration the shell moves from *A* to *B*; the velocities and co-ordinates of the shell are known at *A* but not at *B*. First let us assume that the forces acting on the shell are the same over *AB* as at *A*, i.e. the accelerations over *AB* are the same as at *A*. These accelerations are evaluated from the equations of motion, since the velocity of the shell and the components of velocity are known at *A* and also the co-ordinates of *A*, upon which alone the acceleration depends. Since the accelerations over *AB* are constant, the changes in the components of velocity over *AB* are given by the products of the corresponding accelerations and the interval of time. Hence the velocities at *B* may be calculated. Next, the changes in the co-ordinates over *AB* may be obtained by multiplying the time interval by the average of the velocities at *A* and *B*, and hence the position of *B* may be calculated. This completes the first stage of the calculation, giving approximate values for the co-ordinates and velocities of the shell at *B*.

The second and final stage in the calculation is a repetition of the first stage but in this case we work with the acceleration over *AB* obtained by taking the average of the accelerations at *A* and at *B*, found in the first stage of the calculation. This completes one step; the process is repeated for the next interval, and so on.

We may gain some idea of the speed of automatic calculating machines as applied to problems of this type. For this particular problem the step concerning *AB* takes about one tenth of a second, using a machine able to perform an arithmetic operation in one thousandth of a second. The calculation of the whole trajectory takes several hundred steps and is completed in a time of the order of half a minute. For very fast machines, this time may be reduced considerably. Indeed, machines operating in what is known as the parallel mode may be able to complete the calculation in only a few seconds. With such machines, if a gun were fired simultaneously with the start of the calculation of the trajectory of the shell fired, the machine would complete the calculation not long after the shell had reached the target, if not before. At any rate even with automatic machines of normal speeds the calculation is made almost as rapidly as the shell would travel over its trajectory. The E.N.I.A.C., which was specially built for this type of computation, solves the problem in half the time the shell takes to traverse the distance from gun to target.

Chapter 20

DIGITAL COMPUTERS AND THE ENGINEER

Crinkle, crinkle, little spar,
Strained beyond the yield-point far,
Up above the world so high;
Boy! I'm glad that I don't fly

ANON

We remember once being told that an Engineer could be defined as a man who can do for half-a-crown what any fool can do for a pound. It is this introduction of the economic motive which sets the keynote to the routine computations of the practising engineer.

Three types of computing problem come within the profession's normal orbit. Firstly, there is the sort of work which is generally classed as research investigation, and which usually involves the application of established physical principles to cases likely to be met in practice—the sort of investigation which often ends with the deduction of approximate formulae or tabulated data for use in common design practice. Secondly, there is the problem of the reduction of large quantities of experimental data—a problem particularly acute in aeronautical research—and again usually leading to information suitable for design use. Thirdly, there is the question of optimum design—the problem which is, in fact, the basic professional concern of the practising engineer. In most cases met in common practice, the mere satisfaction of the technical requirement of a specification can be carried out in an infinite number of ways; the problem really consists in the selection of the most economical design.

This article deals with the third category; the first two types of problem are identical with those met by the applied mathematician and the experimentalist in all fields. The third type is peculiar to engineering design and usually demands a considerable amount of information and at least as much computation as is usual with applied mathematical problems. In fact, the engineer usually resorts to a standard mathematical approach in that section of his problem which lies between the specification of dimensions and loads, and the checking of stresses, voltage gradients, aeroelastic effects,* etc.; his basic problem of the "most economical approach" is much wider.

Design Process

The best way of determining the field in which computers can assist the engineer in this wider task is to outline briefly the process of design as it occurs at the moment. We are here concerned with the sort of engineering problem in which mistakes are expensive and in which careful design can lead to considerable saving in cost—problems, in fact, in which a complete design on paper is expected to lead, with later modifications of detail only, to the finished product. This excludes, of course, many problems in industries such as light current engineering where the "cut-and-try" technique is sufficiently inexpensive.

An engineer then, having his specification clearly in mind, conceives, in broad outline, a scheme which will satisfy it, i.e. he sketches out his structure with its principal dimensions, roughs out his power-distribution scheme, or his distillation tower. It is at this point that his judgment and experience come to the fore—his initial choice of dimensions should, of course, be as close to the most economical design as possible.

The next step is to check the assumed design—Will it satisfy the specification? This may involve the working out of a complete design from the assumed dimensions, and checking that design for heat dissipation in the case of electrical machinery; it may call for the determination of stresses in a structure, of top-and-bottom products in a distillation tower, of angle-time characteristics in the case of synchronous generator instability problems, etc. It is at this point that the engineer usually invokes the methods of applied mathematics—even if his specialist approach disguises the fact.

If the specifications are not met, some of the assumed parameters,† e.g. sizes of members, will be altered accordingly, and the process repeated until a sufficiently close approximation is achieved. Actually, if the computation process is long, it may not be worth while to repeat the computations if checking against the specifications shows that the proposed design errs on the side of safety.

Usually, there is still a large number of parameters at the disposal of the engineer; the alteration of some will not affect the

* Aeroelastic effects are those which involve interactions between aerodynamic loading and structural deformations of an aircraft.

† A dimension which we can vary for the purposes of the problem is known mathematically as a *parameter*.

original scheme, and of others will lead to quite radical modifications. The more routine part of the design procedure now includes the extraction of a cost figure and it is rarely that available information justifies high accuracy here. Now ideally, cost* will be minimized by varying suitably the initial parameters—and the nearness of approach to this ideal is determined by such factors as the importance of the job and the time and cost of carrying out the additional computation.

Similar minimal costing is then carried out on other schemes involving a radical change in the initial parameters such as the use of different materials. The engineer, armed with a full range of estimated costs for these various means of satisfying his specification, is now in a position to decide which design to choose.

This process is, of course, not always followed in its entirety; perhaps the requirements of the problem may be such that alternatives involving radical design changes need not be tried—or time considerations are such that a safe design, though possibly uneconomical, must be accepted.

A flow sheet† may be constructed which will show up the "loop" nature of these processes more clearly. The inner loop, the one in which the techniques of the applied mathematician are usually called upon, is apt to be the one which consumes most time and which, from an engineering point of view, is of little interest. The other loops entail the introduction of more and more judgment, and it is in their execution that knowledge based on past experience is so important.

In more mathematical terminology, we may say that the engineer is seeking to minimize a function (usually the cost) of the various parameters at his disposal—and, as his problem is usually a non-linear one, the speed with which his process converges to a final solution depends very largely on the soundness of his first choice of parameters.

Now, the closeness of approach to the most economical dimensions depends on the importance of the problem, the amount of computation in the inner loop, the fact that, near its minimum value, variation of the dimensions assumed by the engineer will

* Note that, in aircraft design, cost and weight are usually proportional to each other; so that in this case, weight may serve as a very simple cost criterion. Cost usually means capitalized rather than initial cost; the object of the design, however, may not be to minimize the cost. The requirements may be to determine optimum value of some characteristics other than cost (e.g. for military purposes).

† The terminology here is that of the chemical engineer, who represents his large-scale processes in diagrams known as flow sheets which are similar to Figs. 20/1 and 20/2. We call a closed cycle in such a diagram a *loop*.

often make little difference to cost, his confidence in his initial choice of parameters and, perhaps most important of all, the number of man-hours he has at his disposal to carry out the work. Designers

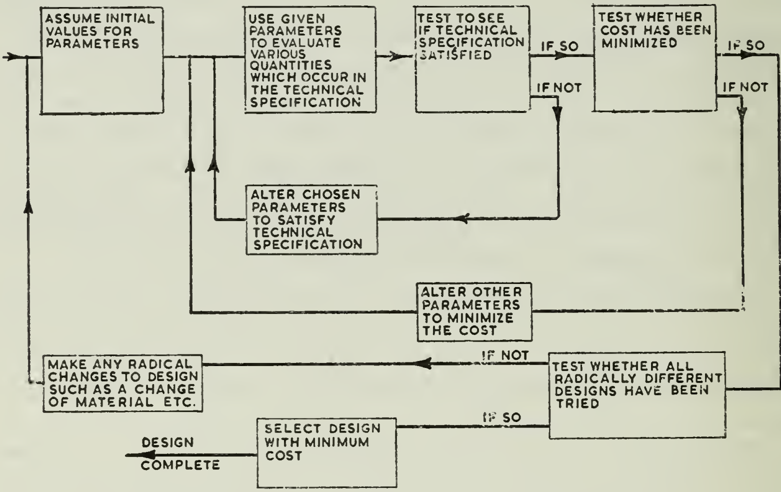


FIG. 20/1. A flow sheet indicating the general design process

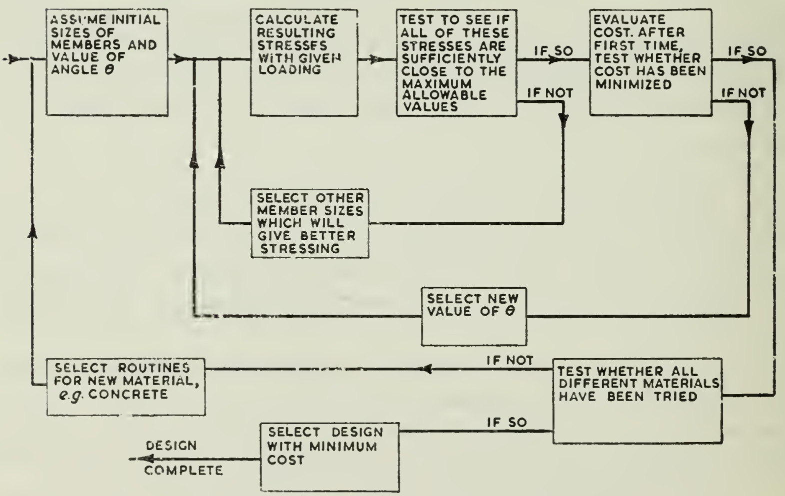


FIG. 20/2. A flow sheet for a simple structural-design problem

are usually busy men, and it is difficult to justify elaborate computations leading to a more refined design when competition for contracts is not keen and when experience has shown that a certain procedure has, in the past, yielded at least a "safe" design.

Factors Necessary for Computing Machine Adaptation

Let us examine the sections of this process at which the use of a digital computer is most justifiable—the justification, of course, being on purely economic grounds. Now, three aspects of a computation in particular fit it for machine adaptation. These are—

(*a*) A large number of repetitive operations, so that the number of times each programme loop is traversed is high. Thus the time saved becomes at least commensurate with the time taken in constructing the programme. Note that this does not necessarily imply lengthy calculations—a loop of a simple programme is also traversed many times if that programme is used sufficiently frequently.

(*b*) The amount of numerical data—empirical data such as tables of sections, numerical coefficients, etc.—should not be too high, otherwise the storage facilities available will be overloaded.

(*c*) The statement of the problem should be such that it can be presented to the machine for solution in an economical amount of time. The interpretation of the adjective "economical" here is important. An economical method for a research student who has constructed his own programme for a particular problem may well not be economical in a design office where a given problem occurs many times in the course of a year and a variety of persons has to be instructed in the presentation of data for machine solution.

Application to the problem of design

In terms of our first requirement, the "inner loop" of the design procedure is usually amenable to mechanization—and this is the section which is certainly most time-consuming. Moreover, most of the processes involved are sufficiently standard to enable "library" programmes devised for other purposes to be adapted with little alteration—a typical example, which occurs frequently in engineering practice,* being the solution of simultaneous equations.

The second loop—the alteration of dimensions to satisfy the stress, aeroelastic, heat dissipation, or other criteria—is usually carried out by readily describable variations of the initial parameters assumed. For example, if we find that a structural member is overstressed, it is made stronger; if an electric motor overheats, its surface area is increased, or the resistance of its windings is decreased. Here the operations involved, although specific to the problem, are often sufficiently simple to warrant their inclusion in a programme—and this is particularly so if a whole range of problems of the same type is being catered for.

* See in this connexion page 396.

It is in this loop particularly, and in the next to some extent, that the question of storage space is likely to become acute. In structural design, for example, it may be desirable to include a table of standard sections in the machine so that heavier or lighter sections may be selected as required. If this cannot be avoided (and usually the accuracy with which we must agree with specifications, maximum stresses, etc., is such that an algebraic representation of the properties of the tabulated data with which we are concerned suffices), we must ensure that our storage capacity is sufficient.*

The extraction of cost data often calls for detailed design of a type best undertaken on a drawing board. However, there is a large field of problems in which sufficiently accurate figures for comparison purposes can be extracted with comparatively little detailed study. Thus it may be sufficient to approximate to the cost of a steel structure by assuming that the cost of the steelwork is proportional to its weight; similarly with transformer design the variable part of the cost may be obtained from the amounts of steel and copper included.

It often happens that this step can be omitted due to the existence of simple relationships between the initial parameters which must be satisfied for minimum cost. In these cases, this loop may be avoided and the number of parameters available for variation is correspondingly reduced.

This leaves us with the outside loop only to consider—the loop involving major design changes. This usually requires such major programme alterations that it is rarely worth arranging to make these changes automatically.

The importance of the third point (*c*)—that of ease of presentation—must be emphasized. Those who have done much engineering computation will be only too well aware that a common source of error in problems involving routine mathematical manipulations (say the solution of simultaneous equations) lies not in solving the equations (where adequate recurring checks are available) but in setting them up. How many times have we discovered to our horror after hours of computation that we have made an initial error in the sign of one of our coefficients?

It is in fact this "setting-up" procedure which has led to the

* It is interesting to note that all relevant data about standard equal-legged angle sections (i.e. L-shaped steel girders) contained in the Dorman Long Handbook (1944 ed.) can be included in less than 250 forty-digit binary-number storage locations. The storage drum of the Manchester University machine, for example, has provision for the storage of 16,384 forty-digit binary numbers.

extreme specialization of many engineers. The work is considerable, and short cuts have been evolved which, though often limited in the range of problems to which they are applicable, certainly help the designer considerably.

For machine computation, however, these special devices of limited application are rarely worth-programming—it is better to use more general approaches even if this means that the machine is going through a proportion of unnecessary operations. The ideal of course is that we should present the machine with a basic minimum of initial information and that it should do the rest; whether this is realized or not depends on economic considerations. Is it worth while expending the time on the necessary programme once and for all—or are we prepared to devote time to instructing intending users in the more complicated system?

Here again, the answer depends on the number of times we expect to use our process and the technical background of persons who will use the routine we are designing. We have always to remember, however, that an upper limit to over-simplification is set by the storage facilities at our disposal.

The earlier points (*a*) and (*b*) are important enough—they imply that the engineer will be placed in a much better position to exercise his broader function instead of having to spend such a large proportion of his time computing—a process in which he has little interest. The last point has two implications—the engineer can avail himself of the latest mathematical methods available on the "black-box" principle, i.e., with knowledge only of the meaning of the input and output data and not of the intermediate processes; and secondly, the relatively unskilled can make use of sophisticated processes of a type which at the moment can only be operated by the most specialized. The effect of this is that the task of manipulating the machine can be left to operators with little knowledge of the processes they are using.

The economies effected by a designer as a result of more accurate information at his disposal result not only from the fact that he can carry his "minimization-of-cost" process much further. Even more important in many cases is the point that more detailed methods of analysis enable factors of safety* to be lowered with confidence—and, when the labour of computation becomes a matter of minor importance, the use of more refined methods becomes feasible.

* A factor of safety of course allows not only for possible defects in material; it also covers the inadequacy of the design methods being used—and as such has often been called a *factor of ignorance*.

An Example. Perhaps many of the general points made above will become clearer if an example of the design of a simple steel structure is discussed in detail.

Suppose the problem is to design a minimum-cost structure of the general shape shown in Fig. 20/3, joints *B* and *C* being assumed stiff and the dimensions *L* and *h* being fixed. The structure is to withstand a sideways force of *P* lb and a downwards force of *Q* lb, and the angle θ is at our disposal. From practical considerations we can

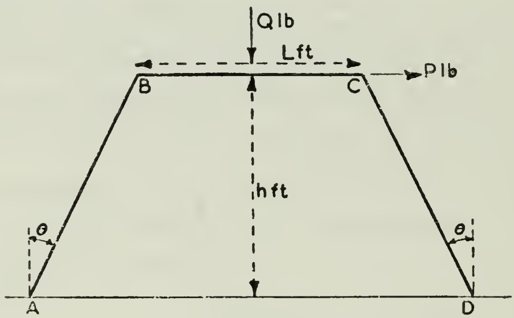


Fig. 20/3

assume that the members *AB*, *BC* and *CD* are of I-section, and of standard size; and for the purposes of this example, cost may be assumed proportional to the weight of steel involved.

Now the non-technical reader must take it for granted that considerations of geometrical continuity and of the equilibrium of the forces involved will lead to the setting up of a system of simultaneous equations, the coefficients of which are functions of the properties of the members *AB*, *BC* and *CD*. The solution of these equations is a standard mathematical process; their setting up can be reduced to a further standard operation,* involving the properties of the individual members and some constants depending on the geometry of the structure.

The solution of the equations can be used, again by means of a process which is perfectly general and can be applied to all structures, to determine the maximum stresses in the individual members, and new section sizes may then be selected so as to give stresses closer to the maximum allowable.

This new selection will invalidate the previous calculation; but several repetitions of the loop should produce a structure in which

* This setting up process, in point of fact, may be carried out as a triple matrix product.[1]

the maximum stresses in all three members are as close to the maximum allowable as is possible.

This structure can then be costed simply by referring to the stored details of I-beams and obtaining the total weight of steel; thereafter, a new value of θ can be tried and the process repeated. A decrease in cost will indicate that we are moving in the right direction—and the utilization of a standard mathematical trick will rapidly bring us to the value of θ which gives a minimum cost structure together with the sections required.

This whole process can be readily handled by an automatic machine; however, if we desire to try the alternative merits of steel and concrete for such a structure, we must modify our programme for concrete structures; and the whole process is most effectively carried out by using this programme as a separate operation. Details are set out in Fig. 20/2.

This particular structure is almost trivial to the structural engineer. However, exactly the same programme is applicable to more complex structures and loadings—all that the machine requires is a geometrical specification of the structure involved in the initial trial, together with the sizes of members which the engineer considers to be reasonable initial values.

Type of Machine Required

The requirements of the engineer may call for the full-time use of a fast machine, or of a slower but cheaper machine in which emphasis is placed on its automatic function rather than its speed; or merely for the hire of a fast machine for such periods as may be required.

The full time use of a fast machine is most likely to be required in a large research establishment where a considerable amount of reduction of experimental data and of fundamental computation concerned with new problems has to be undertaken. It may also be justified in large companies where the machine is to be used not only for engineering computations but also for commercial and accounting purposes. Such a machine would cost at least £50,000 at present market prices, and, on the basis of several estimates carried out for the Manchester University machine and the E.D.S.A.C., would be expected to produce answers to typical engineering problems more than 200 times as fast as errorless and skilled hand computers working entirely by rote.

A slower machine, costing perhaps £10,000 and operating at a speed which is only a few times faster than the skilled computer we envisaged above, will also have its place because of its automatic operation. Such a machine would be at its best when working out routine problems for an organization which has no access to a larger machine but employs perhaps 10 or 20 engineers to do design computation of various standard types. The advantages of the freedom from error which may be expected, even from machines which do make occasional mistakes (since such errors can be eliminated by self-checking programmes), are considerable; and, when we recall that a round-the-clock service can be obtained from a reliable machine, we see that, despite its slowness, the purchase of such a machine may still be justifiable. As cheaper machines usually have a small storage capacity, their usefulness is restricted accordingly.

Hiring a fast computer is certainly most attractive for the small organization, as it removes from the user the problems of maintenance. Whether it is satisfactory or not depends largely on the availability of a machine, not only for the execution of existing programmes, but also for the development of new ones.

At least one person attached to an organization using any of these three approaches should be able to construct programmes or to alter existing ones—once the labour of constructing a library of the more standard routines has been completed. An organization with its own fast computer should operate most effectively with two or three programmers with only a limited mathematical knowledge, and at least one mathematician, skilled in numerical techniques, and preferably with a wide experience of engineering problems.

Problems of maintenance and reliability are not specific to machines used for engineering computations and will not be treated here.* It should be pointed out, however, that numerical checks are as important in automatic work as in manual computing—although, because of the increased speed of computation, they can be spaced farther apart. Errors may arise, not only from machine faults, but also from the occurrence of the unusual case which may invalidate a programme—and the machine itself should be used to detect them.

Possible Improvements in Techniques

We have discussed briefly those features of the engineers' problems which are best adapted to mechanization; it is now of interest to reflect on the changes which will inevitably result from the fullest economical use of computers.

Two main changes are to be expected—changes in the function

* See Chapter 4.

of the engineer and in the quality of his decisions. The second is largely a result of the first, and is, from the lay point of view, the more important.

The first, however, will certainly be welcome to the designing engineer. It means that he will be freed at last from the drudgery of computation. It means that, because obtaining information on which to base his decisions will be so much cheaper, he can afford to work out a greater number of alternatives—a luxury so often denied him at the moment by shortage of time and available effort. It means that, if his programme is arranged properly, he can be sure that his results are as free from errors as the initial data presented to the machine can make them. And lastly, it means that, without having to acquaint himself with the details of a mathematical process, he may make use of it and have complete confidence in the results obtained.

Some Cases Where Digital Computing Has Proved Useful

Some idea of the scope of digital methods can be obtained from the following selection of routine engineering problems which have been solved by digital methods—either with digital computers or (so far) only by punch-card equipment. Problems which are primarily of a research nature are excluded, and the list is by no means complete; nor do the bracketed groups always indicate the only ones who have carried out the problems mentioned.

Structural Problems. The determination of stresses in line structures for a given loading (Cambridge—digital computer).

Flutter and Aircraft-resonance Problems. (I.B.M.—card equipment.)

Stability of Electrical Power Networks after a Temporary Short Circuit. (Cambridge—digital computer.)

Distribution of Electrical Power in Networks. (I.B.M.—card equipment; Manchester (under preparation)—digital computer.)

Cam-design Calculations. (I.B.M.—card equipment.)

Electron-trajectory Problems. (Cambridge—digital computer.)

Conclusion

Automatic computational methods represent an untried avenue of approach for the designer. Let us hope that with the introduction of these methods, he will be placed in a much better position to exercise his proper function.

REFERENCE

*J. Frank*1. Kron, G. *lin Inst.* **238** (Dec., 1944) 399, 443.

Chapter 21

MACHINES IN GOVERNMENT CALCULATIONS

The right time for making changes is when you cannot help it—
THE DUKE OF CAMBRIDGE

FIGURES COLLECTED AND COMPUTED IN GOVERNMENT DEPARTMENTS may differ from those used in other large organizations for several reasons. They may relate to activities which concern only government, or to enquiries so wide in their scope or confidential in character that only government departments or agencies acting on their behalf can provide the effort and incur the expense of conducting them, or to those social, scientific or economic developments which the Government is interested to foster. Government accounting procedures, built on precedent and organized to meet established requirements resulting from dependence on Parliamentary approval, result in forms of presentation very different from the balance sheets of commerce, but the actual calculations and checks involved are similar. At any point, however, in government figure work, and perhaps particularly in government financial accounting, the need to be able to present to Parliament the fullest data available may so influence procedures in computation and the forms of records required to be maintained that the possibilities of using particular mechanized procedures may be greatly prejudiced. Resulting limitations on the benefits which can be derived from the speed and the elasticity of machines as distinct from human labour can be very serious. Economies easily justifiable in commerce although they may involve the loss of information may in matters of national concern have to be evaluated differently. For example, equal trouble may have to be taken in considering large and small sums of money, because the rights and duties of persons are involved, whereas in business it may be wasteful to worry about the smaller items—to pay may be cheaper than to investigate. In all accounting work the need to have a system which makes available records intermediate between the original data and the final answers will affect the way machines can be used; in government accounting these records may have to be more detailed.

Although the benefits which can be obtained from using mechanical aids may be limited in respect of accounting, the possibility of using them for government statistical purposes may not be so much restricted, if only because requirements in the statistical field are frequently changing and forms of presentation may reasonably be varied. The responsibility to Parliament remains, and with it the possibility that data may be demanded which, in a comparable study in a non-government context, would not be expected to be supplied, and this may necessitate the elaboration of procedures and add considerably to the cost of a mechanized system. It is, however, possible before operations are commenced to attach a scale of values to the different items of information and generally to assume that this scale may properly be decided on statistical principles; if there is evidence that a figure will rarely be called for, this is good reason for not having it available.

The kinds of statistical calculations with which governments in advanced countries are concerned range from the utilization of the limited quantitative data which are very often all that is available for the solution of particular problems, to the sophisticated treatment of the precise measurements required in, for example, ballistics or astronomy. Examples of the former type of work are those which fall under the generic heading of econometrics, dealt with in detail elsewhere. The collection and preparation of the original material may be difficult, and much of it, being national in scope, may require extensive computations before the data as finally presented are available, but their subsequent use often involves no more processing than the calculations of simple arithmetical relations. The difficulties here lie in the understanding of the character of the material and of the problems for the solution of which it is being used, not in the computational procedures. Between this kind of statistical work and the mathematical problems posed by his subject to the astronomer and the ballistics expert, etc., are wide fields in which the mathematical procedures are not unduly sophisticated and may be very simple, but in which the administrative requirements and technical statistical problems of a non-mathematical kind—e.g. relating to design of forms, framing of questions and tabulation of results—raise difficulties in preparing for and organizing the necessary computation.

Although in many respects problems of collection, compilation and computation of government statistical material are very similar to those in other large organizations, some differences are evident.

A common characteristic of most government figures is universality; they often concern all members in the country of the groups to which they refer, although there may be different treatment of various classes within the groups. Taking of censuses of population, industrial production and commercial distribution, records of unemployment, changes in employment, sickness, etc., and registration of births and deaths are typical of those activities which increase in scope as society becomes more complex. The need to cover the whole of a particular group may sometimes lead to great elaboration of the information which has to be called for by virtue of the heterogeneity of the various elements in the group and the need to cater for all possible variants. In this kind of situation the bulky forms typified by the Census of Production returns may be required. In other cases universality demands that the information asked for shall be very brief; to ask much would overstrain the ability as well as the patience and truthfulness of the informant and possibly make the task of handling the material well nigh insuperable, and unless the clearest case is made for collecting more elaborate data the demand may well overstrain the patience of the Treasury which has to pay for it. The Census of Population is here typical. Not more than about fifteen items of information are collected in respect of each person apart from those relating to the household as a whole but, as the 1951 Census will provide information relating to nearly 44 million persons in England and Wales, and as the different items of information will contribute to a very large number of tabulations, the labour of handling the information derived from even a limited number of questions will be enormous.

Although by the nature of this census the number of questions which can be asked is limited and the number of associated facts available without questioning is small, the total use of the data produces very elaborate material. In 1921 there were only 13 columns on the census form, which was regarded as being up to the limit of the public capacity; nevertheless, the answers in these columns, together with particulars of the place in which the census was taken, etc., and the special enquiry about persons speaking the Welsh language, resulted in 92 tables, many of them covering a number of foolscap pages of close print.

In that year the census was taken on the 19th June, 1921, and publication began with the first volume of the County Part series in October, 1922; further volumes appeared up to July, 1925 and the General Report was completed in 1927. Full information therefore only became available within about 4 years of the time when the next census was due. The 1931 census was more restricted than that of 1921—among other reasons because of the possibility that under the Census Act of 1920 another Census might be taken at the end of five years—but in spite of this the analysis required each of 39,952,377 cards to be passed through the machines 16 times on the average—even after allowing for these passages which enabled cards to be both sorted and counted at the same time—and was not completed when the war broke out in 1939. The final report, although largely written before the war, was in fact not published until 1950. In spite of the long time which large-scale census-taking requires before complete information can be published, it is rare that the analysis of the data is exhaustive; the limit is set by cost. If a system more economical in cost becomes available more can be done; if it is quick the usefulness of the additional information will be very greatly increased.

The method of analysis in which the original data are transferred to punched cards seems unlikely to be varied in the very near future. The analysis of some millions of cards can in total take a great deal of time, in spite of the fact that they can be passed through the machines at the rate of 24,000, or even 40,000, per hour. Electronic machines should, however, be able to read and count cards, classifying them simultaneously under several different headings. The memory of the machine can be used to accumulate the figures which are being tabulated, each figure as it is read being added to the several totals to which it contributes, so that when the data have all been passed into the computer, the figures for the various totals are stored in the memory and are immediately available. As computers can modify their own instructions according to the results they obtain from calculations, systems making use of them should show advantages compared with those which increase considerably in cost unless a final decision can be given as to the procedures to be followed before the machines are called upon to commence the sorting and counting.

In census work, as in all analysis of a number of factors relating to individuals, an important preliminary operation is the scrutiny of the original documents for obvious errors and for inconsistencies which may not be immediately obvious. (Although enumerators may at the time of collection take steps to ensure that inconsistencies are avoided, subsequent coding errors may introduce them, and not all work of this kind can be handled with the elaborate precautions taken by the Registrar-General.) The analysis would show some of the errors to have been present—for example in census-taking where dates of birth have been wrongly given, or in those studies in which the absence of penalties is a temptation to the practical joker—but before mechanical operations can be successfully used it is necessary to ensure that there are not in the data inconsistencies which will lead to tabulations being prepared in which the totals of various rows and columns which ought to check one another do not do so, so introducing doubt about the accuracy of the machine. A satisfactory method for removing inconsistencies is essential, and when a mechanical method is available—for example, to ensure that recorded dates of birth, marriage and births of children are not incompatible —it can save a great deal of clerical work of a kind which is particularly tiresome because it demands a high degree of concentration without having any intrinsic interest.

Although much of the information from the census of population is immediately of practical value without further processing, the figures serve also to provide denominators for a range of calculations of proportions to provide valid vital statistics. A single example may be cited, taken from the Registrar-General's *Statistical Review* for the year 1939. In that year the usual procedure of estimating populations for each area at the middle of the year was extended by the use of the National Register of 29th September. To make comparisons of mortality in the various towns and areas of the country, deaths registered in each area had first to be converted into crude death rates (registration of persons dying away from their place of registration being transferred). Since these crude rates are affected by the sex and age distribution of the population of the area an "area comparability factor," calculated for each area to make the appropriate correction, had to be applied. These factors, recomputed from the National Register data, when multiplied by the crude death rates, give for each area a locally adjusted death rate which suffices to standardize, for the purpose of valid comparison, the mortalities of different areas *in the same period of time*. The procedure does *not* make possible valid comparison of mortalities at different periods of time. For this an analogous "time comparability factor" must be introduced to remove the effect of sex-age changes in population taking place from time to time; when the product of the two factors is applied to the crude death rate a completely standardized local death rate is produced and, within the limits set by the assumptions made in the calculations of the factors, comparisons between areas and at different times can be made.

The calculations had to be made in respect of each of over 1,500 areas, and the extent of the computations serves to illustrate why, in many contexts, comparisons of only crude figures are made. The danger of this is that false conclusions may be drawn. The proper "standardizing" or other preliminary treatment of figures which have to be used for many commercial and administrative purposes before they can safely be used to make valid comparisons, is liable to take too much money, time and patience. Many computations only become possible in limited time if appropriate machinery is available; many are still impossible because appropriate machines are not available.

Different procedures are needed when industrial and commercial data have to be analysed—statistics of production of industrial products are often only meaningful if they provide analyses by detailed categories of types and grades. While, therefore, the number of establishments from which figures have to be collected may, in a particular trade or for a given product, be quite restricted, especially if, as is often the practice, small producers can be left out of the enquiry or treated separately, the data required from each may be very elaborate and the total number of calculations very large. Mechanical aids are called upon to prepare and store the detailed information so that summarized data may be produced quickly. An important cause of complaint in these studies is that details are produced first and summaries later, when the latter are required first. The heterogeneity of the material may make it impossible, and would certainly make it dangerous other than in those fields where careful experiments have been made, to put too much reliance on economies which can be derived from sampling procedures. In population work, analysis of a very small proportion—say one per cent—of the whole may produce many accurate national figures, but production statistics do not relate to large homogeneous groups, and the establishments from which they are collected show even less homogeneity. Information required must wait on the speed of the system used.

Preliminary figures from the Census of Production for the year 1935 were published in 1937; final reports were issued from 1938 on, but were not completed when the war came; the data collected from the Import Duties Enquiry in respect of the year 1937 (similar in content though less extensive in scope than those of the Census of Production) were not published—except for some of the preliminary reports issued in 1939—before the outbreak of the war.

In this class of work, however, there are other advantages to be gained from machines working to predetermined rules, both in analysing data to obtain more meaningful conclusions, and in pursuing the economies which might be expected to follow from a sampling system adapted to the class of work.

Classification of businesses, whether industrial establishments or commercial concerns, is necessarily made in the first instance according to familiar customary descriptions. These classifications, based on verbal associations, are sometimes insufficient to separate the economic characteristics of various groups. They often fail to distinguish between manufacturers using processes so different from one another that they really belong to different trades; they do not without further classification make the important distinctions depending on size of establishment. Important sub-groups of establishments within particular classificatory groups may only be identifiable from the statistics themselves since the criteria leading to identification may not be known beforehand; it is part of the enquiry to discover them. The calculation of important secondary statistics may therefore have to be preceded by extensive analysis for which a full programme cannot be laid down in advance. For example, the value of an analysis by size of unit will depend on the number of units in the size groups, and the calls on the analytical machine will not be known until the preliminary analysis is complete. If a wide range of qualities of a product is made there may be a tendency for specialization which might affect, for example, calculations of net output per man, and averages which take no account of significant quality or other differences may be dangerously misleading and may indeed yield trends quite different from those of the constituent parts. In these circumstances analysis may require preliminary calculations from the forms (say, of average prices per unit) in order that firms exhibiting important differences in their figures may be distinguished. (This is also typical of one of the crude tests commonly applied to figures to ensure that correct data has been supplied, since errors in copying often produce figures which can by this method be shown to be absurd.) If the computing system can carry out a programme of calculations of average prices per unit of product, subsequently sort the primary statistical material into groups determined by these average prices, and calculate secondary statistics for the several groups separately, information may be disclosed and suggestions for subsequent investigation appear which would not be evident from averages for the trade classification as a whole. This example of classification by average price per unit is only one of a number of possible systems which might usefully be tried, since one of the objects of research into census data is to examine the possible relationship affecting significant secondary statistics. Size of business, location of business and inter-relationship between different types of business may appropriately be examined. From these calculations the necessary framework of groups in the whole system may be determined so that a stratified sample may be prepared. In this difficult field the economies obtained by sampling may never be very great; certainly sampling to give valid results would always need a very careful system of stratification of data to be decided. But, apart from its possible use in designing a sample, stratification is a necessary part of the analytical procedure.

Index Numbers

As index numbers were introduced, and are used in all advanced countries, to measure and present economic and sociological characteristics relating to the country as a whole in a simple form, their preparation is commonly a government responsibility, and a wide variety of indices, both specialized and general, are now prepared by the governments of this and other countries. Many famous series have been prepared by private persons and organizations, but for official use it is usually desirable for governments to have official series available, and in many cases the cost and labour of compiling the data can often only be faced by a government organization.

Most indices have in the past been calculated from relatively limited data. Famous index numbers of wholesale prices, for example, have depended on a quite limited range of quotations; Sauerbeck's index with 54 items, and the Board of Trade Index of Wholesale Prices with about 200, were both compiled on the assumption that, if carefully chosen, quite few price series would measure the movement of the average of prices, or in its reciprocal, the value of money.

The computations have therefore not been very elaborate, and even in periods when price movements have been rapid, it has been common for all work to be done with no aids beyond those which manually-operated calculating machines could supply. It has not been the common practice to regard the price movements as a sample from a universe of movements and to attach to each a statistical measure of its accuracy. When the work is regarded in this way, and particularly if it is also extended to give much more detailed coverage of the economic groups between which price movements may show a substantial degree of independence, the total amount of work becomes considerable. The Wholesale Price Index Number published in the United States, for example, is based on prices of nearly 900 major commodities, and is only one of a number of price indices prepared from primary market-price data.

The various stages involved in computing most indices may conveniently be illustrated by the calculation of index numbers of prices. Initial information for the periodical (usually weekly or monthly) computations is simply a report of prices of selected representative, and clearly distinguishable, commodities. In the type of formula most convenient to use and at the present time probably most commonly used, each of these prices has to be related to a corresponding price in the base period, yielding a proportionate change, to which is attached a (usually constant) multiplier (the "weight" attributed to a particular commodity). The series of resulting products then has to be aggregated and related to the corresponding aggregate in the base period. The further stage of calculation commonly required in index-number work, after the accuracy of the figure has been determined, is to discover which of the various elements comprising it are most responsible for the particular movement which has been calculated. This necessitates the individual movements from various commodities being calculated and multiplied by their weights, and the listing of those which on some pre-determined criteria are considered to be significant. This again requires a great deal of calculation to be done while very little recording is required, since only the most significant items need to be reported. Index-number computations are therefore in many ways typical instances of those calculations in which a relatively small amount of data fed into the machine results in a big range of subsequent calculations. Modern electronic digital computers which could be programmed to undertake calculations of this type entirely automatically and at great speed should do much to facilitate these processes.

Index numbers have to be computed for many other purposes. Indices of wages are calculated monthly, of earnings at six-monthly intervals; import and export price indices are prepared at monthly intervals and average value indices (which take into account changes in relative quantities of goods entering and leaving the country) are calculated quarterly; indices of the quantity of production, in total and for various industries, are published monthly, indices of retail trade are prepared monthly; and so on. They usually involve somewhat similar procedures; calculation of certain variables, the operation on these variables by pre-determined multipliers and the analysis of the resulting movements in the figures to try to determine the important influences which cause them.

Trade Figures

The preparation of statistics of overseas trade, involving the recording of weights and quantities, and places of origin and destination of goods, require mechanized aids in the handling of a very large number of original items in the shortest possible time. As usually compiled they require a great deal of material to be fed into the system and relatively few operations to be performed on each item. The heterogeneity of the data has resulted in much of the work of statisticians engaged in summarizing and analysis being devoted to systems of classification. These systems of classification are attempts to meet different needs and there are obvious administrative difficulties in providing the range of individual figures and varied summaries which all users would wish to have. The type of machine required to do the work satisfactorily, as in many kinds of work concerned with economic and commercial data, would have to have a very big memory and be capable of absorbing rapidly a big quantity of data.

So far, nothing has been said about the uses of machines for the handling of large quantities of data relating to individuals or particular articles. It is to be expected that, in the future, the convenience of electronic devices in making records of information concerning large numbers of persons will be widely used in government social services. Records of insurance are needed to be available in detail at short notice as well as to serve in the provision of statistics for both administration and research. There will be a growing need for histories of ill-health to be fully available, and perhaps also of education, training and employment. The very large amount of data which have to be handled in this class of work may call for modifications in existing machines, but the use of electronic methods will be the obvious way of handling work of this kind.

Statistics in Military Administration

The army in war was about as big as the population of Norway and its authorities accepted responsibility for the details of the daily life of its members much greater than any western country has ever pretended to take for its civil population. Each of the service departments has need for elaborate computations in planning and carrying out operations. New instruments of war called for completely new calculations; for example, in the mapping of air raids to overcome the difficulties created by the absence of continuity in the triangulations of England and Germany.* In administrative matters the Army probably had the greatest difficulties: it was larger, and by its character had to accept recruits more variable in quality than did the other two services; its functions in operations were more varied and it had the principal responsibility in occupation; when joint services were required the Army generally had to provide them on the "biggest-user" principle.

In war military records are so difficult to establish and to preserve that continual complaints have led to frequent efforts to reduce them, efforts often successful simply because there is so much material to attack. In respect of individual soldiers, however, access to a full personal record is one of the necessities in maintaining morale, while much detail is now needed in administration; the demands made in providing, servicing and replacing the complex material of a modern army makes the organization rather than the reduction of data the primary need.

Opportunities for the use of elaborate kinds of computing machinery arise from the full records of both men and equipment; the need for them from the many calculations which are required. Civil organizations are usually in contact with their staffs for not more than one-third of each day and do not expect to have to impose on them a strain comparable to that put on a soldier in war. They may have complicated plants and many vehicles and may have information comparable to the army records about each one of them, but they have not such opportunities as an Army for studying quantitatively the results of experience. Army equipment is used under a wide variety of conditions, and it is therefore possible to plan statistical experiments to test the probable values of new equipments and assess the results of new operating procedures.

Conditions for using mechanical aids are, however, very unfavourable. Information initiated in field conditions, whether within the battle area or outside it, must be recorded on simple forms; there may be so many occasions which forbid the use of special systems

* The radio stations which directed the R.A.F. over the Ruhr were sited in England. The ordnance survey of Germany did not use the same geographical "grid system" as the survey of England, and it was necessary to recompute the German maps in order to join them to the British survey system.

and elaborate forms designed to serve machines more efficiently that their introduction is prohibited. The sources of data are mobile and communications sometimes difficult to use, or they may even have been destroyed, so that the time and distance between these sources and the machines are very variable. The uncertainty of power supplies may make it difficult to bring machines near to the point of origin of the data to be fed to them or for the resulting statistics to be supplied to their principal users.

Nor can a plan always be laid down as to requirements. These vary as the technique of war develops and as geographical areas and administrative organizations alter. A change in conditions may affect, for example, the interchangeability of surpluses and deficiencies of weapons or supplies, very quickly widening or narrowing the area over which this is possible within a particular formation.

The essential speed in operating a useful system is unlikely to be obtained only by improvements in machines. The important delays in the system usually occur before information is supplied to the machines. In this kind of work it is likely that progress will best be made by establishing a dual service; first a fast news service to supply minimum needs with the simplest operations and most elementary equipment, and secondly an elaborate mechanized system which will provide, after some delay, the full data required. The second serves not only to correct and complete the first, but also to improve the guesswork inherent in it.

Similar difficulties may arise whenever machines are used in large administrative systems. They require all parts of processes to be fully organized. Until this is done they can be embarrassing in revealing defects in definitions and classifications which can be glossed over in preparing summarized returns or "absorbed" in giving "average" results. Perhaps one of the most important services to be obtained from machines which do elaborate computations will prove to be the revealing of skeletons in organizational cupboards.

Chapter 22

THE APPLICATION OF DIGITAL COMPUTERS TO BUSINESS AND COMMERCE

Be not frighted; trade could not be managed by those who manage it, if it had much difficulty—DR. JOHNSON.

IT IS UNDOUBTEDLY TRUE to say that far more arithmetic is done every day in keeping the accounts of businesses, large and small, than is done in all the scientific and mathematical laboratories of the world. Calculating machines, such as adding machines and punch-card equipment, were all developed primarily for commercial work, and the late Dr. Comrie devoted his life to exploiting these standard machines in scientific and mathematical computations. The high-speed automatic digital computers which have been described in this book were all designed originally to solve scientific problems; they will do as much arithmetic in a week as most men can do in a lifetime, and it seems reasonable to inquire whether they in their turn can be adapted to reduce the ever growing burdens which the elaborate organization of modern business throws on the white-collar worker.

The idea of mechanizing accountancy is very old. A system of recording the receipt of money by making notches on tally sticks was introduced to the Exchequer by William the Conqueror. The system was in use until it was abolished by Parliament in 1782, but it was not finally discontinued until the death of the last of the Chamberlains in 1826. The tallies were stored in the Star Chamber, which was filled to overflowing, and when in 1834 it was necessary to use the room for other purposes, they were ordered to be burnt, as by then some revolutionary spirit had decided that since pens, ink, and paper were available it was no longer necessary to adhere to the traditional routine which treated these notched sticks of elm as if they were pillars of the constitution. One would have thought that the sticks might have been used for fuel by the poor of the neighbourhood, but official regulations required that they should be treated as "secret waste" and confidentially burnt within the Palace of Westminster. The stoves which were then in use in both Houses of Parliament were notoriously inefficient, and seemed to have been contrived to give Members and Peers hot heads and cold feet. One afternoon in October, 1834, while the tally sticks were being burnt, a party of visitors noticed that the House of Lords was pleasantly warmer than usual. An hour or two later the whole place burst into flames, and both Houses of Parliament were burnt to the ground. The tally sticks helped to make an enormous fire, which lasted for nine hours and was the biggest blaze seen in London between the Great Fire of September, 1666, when Mr. Pepys saw "Saint Pauls burnt out and all Cheapside," and the fire raids in December, 1940. The firemen were directed by the Prime Minister (Lord Melbourne) himself, and so many people came to watch that three regiments of Guards had to be called out to control the crowds.

It seems improbable that any other system of accountancy will ever have such a profound effect on the structure and the machinery of government, but this incident does demonstrate the importance of a reliable system for "declassifying" secret documents and for disposing of secret waste.

There is no doubt that some drastically new approach to the problems of large-scale administration is long overdue. It is notorious that the amount of paper work which has to be done increases year by year, but few realize how monstrous the load has become. The 1951 census disclosed the startling fact that 2,100,000 people, or ten workers out of every hundred in Great Britain, are engaged in some kind of clerical work. The number has increased by 67 per cent since 1931—this is in fact the fastest growing of all professions—and we are rapidly becoming a nation of clerks. Most factories are elaborately mechanized, and their output has increased correspondingly, but the number of people engaged in administration has increased almost threefold since 1920. Since 1940 the salaries which clerical workers earn have increased proportionately less than those of many manual workers and engineers. Furthermore many clerks are working in very large offices; there are almost 10,000 in the Ministry of National Insurance in Newcastle for example, and more than 19,000 in the headquarters of the Metropolitan Insurance Company of New York. There are not many factories in the world as big as either of these offices, which are wholly engaged in making marks on paper. It is probably true to say that some of the dullest repetitive routine work that is done in the world today falls to the lot of white-collar workers in business offices and government departments.

The past few years have seen a rapid growth in the development and introduction of office machinery of all kinds. Punch-card machines, comptometers and automatic addressing machines, for example, are very widely used—modern business would scarcely be possible without them—but useful though these machines are, they are unable to do very much without detailed guidance by their operators at every step in the computation. Digital computers can follow out an elaborate procedure quite automatically and can guide themselves through it by "choosing" according to prescribed criteria. It is to be expected therefore that they could take over much of the work which now occupies so many thousands of clerks.

The very size and complexity of the big offices means of course that the majority of the clerks in them have to become mere automata—cogs in a giant machine. The clerical operations in a big office have to be organized in as much detail as the mass-production lines in a big factory. In a small office there is room for the initiative and imagination of the individual, but if many of the 19,000 clerks in an insurance office attempted to use their imagination they would wreck the system completely. The routine calculations are broken down into simple standard operations, and if the planning has been done efficiently it is seldom necessary for the clerks to refer to senior executives for the elucidation of doubtful points. Perhaps this very fact implies that some of these giant organizations already have within them the seeds of their own decay. The more completely the routine can be split up into small separable operations, each of which can be entrusted to a human "automaton," the more likely it is that the whole thing can be mechanized. Digital computers will really be useful in business only when they can make nearly all decisions for themselves. This is already possible far more often than one might at first think. Many human judgments are based upon an imperfectly remembered series of numerical criteria, and they would probably be much improved if they were more strictly related to these numbers. Machines are capable of following out a very complicated chain of reasoning if it can be reduced to numerical terms or if it can be accomplished by the application of formal logic. In so far, therefore, as office work can be reduced to standard "rule-of-thumb" processes, however complicated they may be, a digital computer will be able to handle it.

Although accounting and clerical procedures differ from firm to firm, the differences are almost certainly less than those between the problem of calculating a firing table for a gun and an investigation of the stability of the main electricity grid system. Both these computations, as we have already explained, can be easily and efficiently undertaken by the same digital computer. In every office clerical work involves filing and sorting data such as that on bills, on receipts, on orders and on salary cheques, as well as a certain amount of arithmetic. It has long been realized that clerks who are engaged in only one operation perform it more efficiently than those who have to do several different types of computation during the course of a day. About a hundred and fifty years ago both the French and the English governments undertook the preparation of mathematical tables. The work was so organized that most of it was reduced to simple addition and subtraction. It was found in both countries that this work was most reliably done by computers who had never learnt anything else, and could neither divide nor multiply. Similar experience made British railway companies refuse, until 1870, to employ in their signal boxes any man who knew how to read. Illiterate men had better memories and were more efficient, but they were hard to find after the introduction of compulsory universal education, so the railway companies had to develop a simpler signalling system which could be operated by men who had learnt to read.

Nowadays simple arithmetic can be done reliably enough on standard desk machines, but the system which is needed to ensure that all the papers needed by a clerk in order to do a particular sum arrive simultaneously at his desk is often more difficult to organize and more expensive to run than that part of the office which is concerned with the operation of the calculating machines themselves. A modern digital computer should be able to read almost instantaneously any of the information stored in its memory, and to route it into its mill, and it may well be that the ability of these machines to organize the data which they contain will prove to be more important in office work than the high speed at which they can do arithmetic. Nevertheless it is worth reminding ourselves than an ordinary clerk can do about 500 multiplications, each of one ten-digit number by another, in an eight-hour working day, or he can add up about 1,000 such numbers in the same time. An electronic digital computer does as much work in a second or two.

Our discussions so far have been couched in the most general terms, and it is time to introduce a note of realism by describing a few typical problems which have been solved in the past year.

To the fortunate majority who are not engaged in it themselves, office work is a great mystery almost entirely surrounded by shorthand typists, so we shall discuss in turn some representative computations which do in fact seem to be of importance and which must be done somehow or other at the present time. We shall not attempt to discuss anything so complicated as the mechanization of an insurance office; this problem has been discussed by the North American Society of Actuaries who have published a lengthy report on the subject.(1) We shall confine our attention to a few simpler problems of which we have direct experience. So far there has been little experience in this country of the application of digital computers to office work. One machine, which is in all essentials a copy of the E.D.S.A.C., has been built specially for the purpose, but it has not yet been brought into use in commercial work and it is too soon to be sure whether the aspirations of its builders will be fulfilled. All the other machines which have been made so far are installed in universities or research establishments, so that there has not yet been an opportunity for comprehensive trials of a digital computer in an ordinary business. Nevertheless the whole subject is being intensively studied in several places both in this country and in America, so in this chapter we have attempted to make some forecast of possible future developments as well as to describe some of the work being done in Manchester at the present time.

A problem which appears to be very suitable for solution on the machine has been propounded by a firm of market-research specialists.

Their "field" men collect details of annual and bi-monthly turnover and sales and purchases of individual commodities from more than 800 food and drug shops. The data are recorded in detail on forms which are then sent to headquarters for processing. The shops have been carefully selected in order to represent a fair sample of all the retail shops in the country, but in order to prepare from the available figures a reliable estimate of the sales for the whole country, and the way in which the sales of each product are distributed among the various types of shop, it is necessary to subdivide the sample shops into twenty-one different categories, according to the district in which they are situated, the size of city and the type and size of shop. The data must thus be sorted under different headings, each of which covers a whole range of products. It is then necessary to multiply the sales in each class of shop by weighting factors, which have been elaborately computed from a study of the best available figures on the retail trade of the country; finally totals are accumulated to determine sales.

At present several hundred clerks are engaged in this routine work. They make use of desk calculating machines and an elaborate punch-card installation. It would be a great help to the firm if the entire operation could be taken over by a digital computer. The machine would probably reduce costs, it should certainly reduce the chance of error, and most important of all, it would speed up the preparation of the company's reports so that their customers could more quickly discover the effect of changes in their sales policy.

A programme has been written for the Manchester machine; it has proved to be most convenient to write all the information for a given product group into the drum at once so that the machine can process it quickly and easily. Further experiments are in progress, and it seems probable that the computer will handle this work very efficiently.

Another problem to be considered is one of production planning in a large factory making such products as chemicals, soaps, cosmetics, drugs, etc.

The manufacturing process starts with certain basic ingredients A_1, A_2, A_3 . . . which are mixed together in definite proportions to make intermediate products B_1, B_2, B_3 . . . . These substances in turn are again mixed to produce C_1, C_2, C_3 . . . and so on. Let us suppose that the final product P is made by mixing together the C's. The manufacturer's problem is this. He wishes to make several final products P, how much of the basic raw materials does he need, and how much by-product does he make? In practice, of course, the problem is complicated by the fact that the end-product may contain some of the A's, the B's and the C's, and moreover, some of the C's may be mixed with the A's to make more B's.

This computation can be set up formally as an example of matrix multiplication, and a programme can be written for a digital computer. The machine is very well suited to handle it, as there is a great deal of repetition and the same type of computation is applied to many sets of numbers; moreover the relations between the A's, B's, C's and P's are constants of the manufacturing process and can be stored in the machine or kept on punched tape ready to be fed in when required. Once the programme has been designed it will handle any problem which arises in practice.

We have made a statistical analysis of the effects on the finished product of changes in some of the raw materials and in the condition of operation of a big factory. In order to do this it has been necessary to develop a programme to compute correlation coefficients and the sums of various powers of a large number of variables. This, too, is a long and tedious computation which is admirably suited to a digital computer. The method by which this calculation has been done is shown diagrammatically in Chart III.

The problem of organizing the factory so as to make the best possible use of the available machines is an example of the "linear-programming" technique which is discussed in Chapter 23.

All these computations are of special interest only to a few, so for our last example, with which we shall deal in more detail, we have chosen the routine computation of wages and the elaborate procedure which is necessary to cope with "pay-as-you-earn" income-tax deductions.

We have investigated how we could work out the wages for a factory employing about 4,000 people who are making light electrical equipment of all kinds. While the computer is preparing the pay slips it is required simultaneously to produce all the information which will be needed by the cost accountants.

In the factory under consideration, there is a bonus scheme in operation, and three-quarters of the employees are in this scheme. Each man needs, on the average, nine different job sheets and a clock card every week, these documents contain between them more than 500 decimal digits.

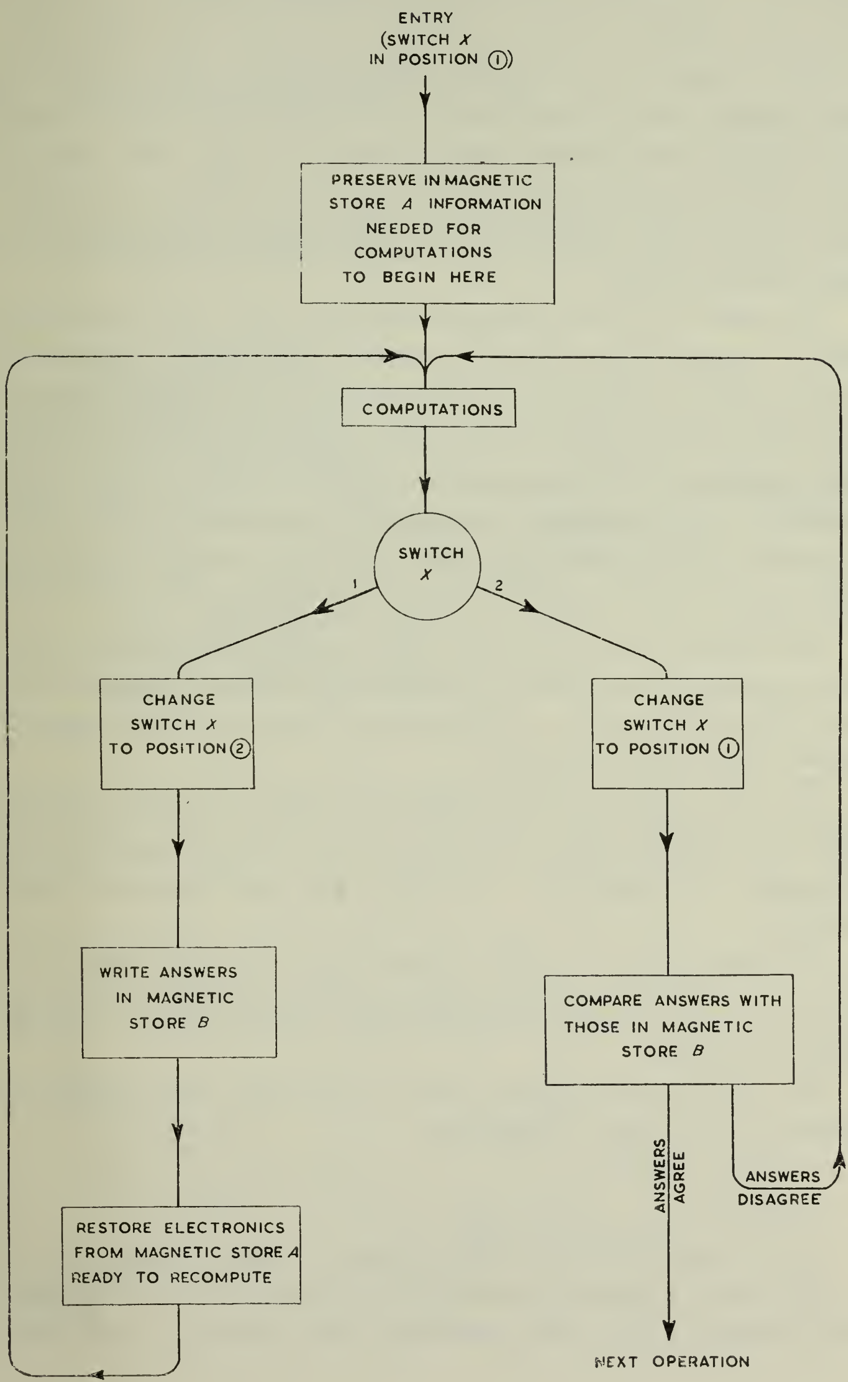
The wages department works out the standard P.A.Y.E. deductions and such things as holiday funds and savings schemes. About 1,000 man-hours are spent every week in preparing the pay slips and in doing associated work.

The Manchester University machine can do the computations themselves in a little over an hour, but it is necessary to consider both the time involved in getting the information into the machine and that required for preparing the pay slips and other output requirements.

Before the machine can start work, it has to be supplied with all the data on the clock cards and work sheets, which have to be transcribed on to punched cards or tape. About 35,000 cards are needed and these take about 150 man-hours to prepare, and several hours to feed into the machine.

The data which have to be kept from week to week, such as rates of pay, total wages earned, etc., amount to about 400 binary digits per man. This underlines the desirability of adjusting existing

CHART I. DIAGRAM SHOWING A METHOD OF CHECKING COMPUTATIONS



systems to fit in with the use of the computer, since it is probable that with a slight reduction of output information the storage figure could be reduced to about 250 digits per man. This information could be stored on the magnetic drum, which is very suitable for the purpose, since it would not lose any part of it during the week (its memory is permanent) and, of course, the data which have to be carried forward until the following week could be written over those brought forward from the previous week on the same drum. All obsolete data would thus be automatically erased as soon as they had been used.

In order to give the reader some idea of the work involved in such a computation, which by normal standards of commercial work is quite a simple one, we have given further details of part of it, namely the calculation of net wages from gross wages. The flow sheet is given in Chart II (shown opposite) and is more or less self-explanatory. The computer is supplied with the necessary data relating to each employee, including his gross wages, and it then computes all the deductions which have to be made for tax, for National Insurance contributions, for holiday club and savings subscriptions, etc. Finally it prints out the pay slip.

We assume in the programme that some of the input data have been produced in the form of punched tape by the computer at the end of the previous week. These data are written on to the magnetic drum when they are required.

The only additional permanent data which are required are the figures for income-tax Table A for the last week of the year; this table gives the provisional allowances, other than earned-income allowance, in respect of a full year for each income-tax code number.

The first operation of the computer is to take this Table A for the 52nd week and to compute from it Table A for the current week and any other necessary constants which depend upon the specific week of the year.

The data for the current week are fed into the machine by means of a data tape. This shows—

1. Works Number.
2. Name.
3. Gross Wages for the week.
4. An indication if it is a new employee's first pay day.

The works number is treated as a code number which identifies the location in which the permanent data relating to each man

CHART II. WAGE DEDUCTIONS

The process of computing typical wage deductions is illustrated. Table A for Week 52 and personal data carried forward from the previous week are written into the magnetic store before the calculation begins. Three deductions are shown here—

1. P.A.Y.E. Income Tax.

Table A for the current week is computed from Table A of Week 52. Various constants required in the calculation for the week are also computed before the data for separate individuals are considered. The employee's code number, which is kept with the rest of his personal data in a compact form, is used to extract his tax-free income to date from Table A. His total taxable income is found by subtracting from his gross wage to date in pence and the remainder of the income-tax calculation is in shillings. Int. $|X|$ signifies the integral part of X.

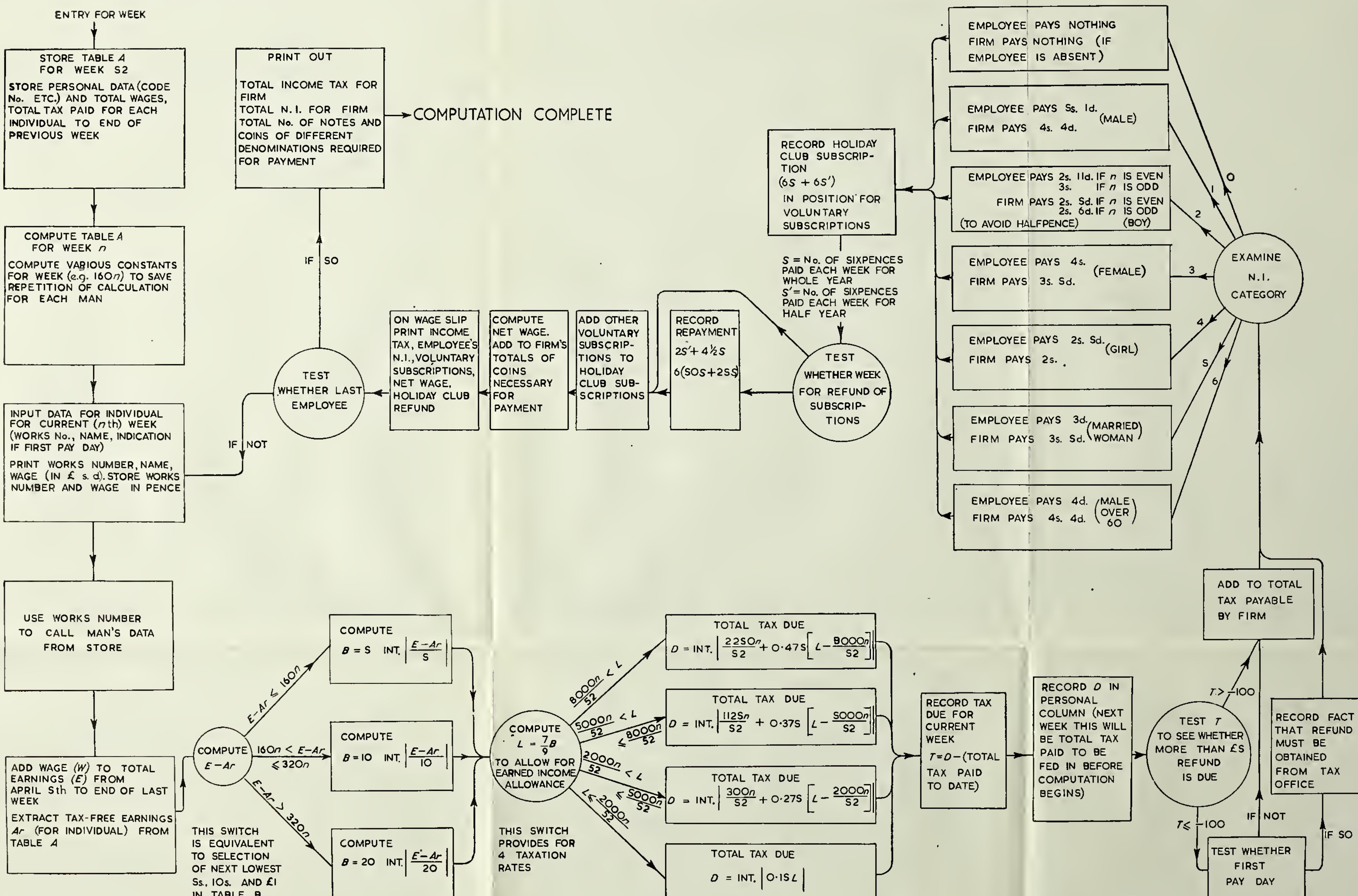
2. National Insurance.

The employee's National Insurance category is stored and this selects the appropriate contributions.

3. Voluntary Subscriptions.

(*a*) Holiday Savings Club. Employees may contribute any fixed number of sixpences per week up to 20s., and are refunded the week before the works holiday. Contributions may be begun immediately after the works holiday or six months later. The interest on money paid for the whole year is 4½d. for every 6d. of the weekly rate, and that on money paid for the half-year is 2d. for every 6d. of the weekly rate.

(*b*) Other Subscriptions. These may consist of contributions to any of the following: Foremen's Benefit, Oldham Hospital, Salford Hospital, National Savings, Recreation Club.



have to be stored. The wage for the current week is added to the total at the end of the previous week, tax-free earnings are extracted from Table A, and the difference is computed. The machine has to do three different kinds of computation according to whether, as is shown in Table B, taxable income is to be taken to the next lower 5s., 10s., or £1 unit. The course of the calculation is shown on the diagram.

The computer then deducts the appropriate proportion to give the earned-income allowance, and proceeds to compute the total amount of tax payable to date. To do this it has to split the total income into the separate amounts on which different rates of tax are payable. The total tax paid to date is then stored.

At this point the machine has to discover whether a refund is due and, if so, whether this refund exceeds £5, and again, if this is so, whether it occurs in the man's first pay week, since in this case any refund may not be made by the firm but is given by the Inspector of Taxes. The fact that a refund is due has to be printed out. The machine then makes the appropriate deductions for National Insurance, recreation club subscriptions, holiday fund, etc., finally testing whether or not the case with which it has just dealt is the last employee. If it is not the last employee the computer returns to the beginning and repeats the computation for the next employee; if it is the last employee the machine computes the totals which are required by the firm, i.e. total income tax payable, and total national insurance; the final computation provides the total number of notes and coins of different denominations which are required for making up the pay envelopes. The calculations occupy the computer for about half a second per employee, so that by using pre-printed slips and a fast printer it is possible to prepare about 3,500 pay slips an hour.

We have also investigated how the machine could be made to analyse manufacturing costs in the same factory, for the management are without much of the detailed information they would like to have. Has any business man anywhere ever really known what it cost to make anything?

The necessary data on clock cards and work sheets which describe the operations of 1,500 workmen during the year correspond to about 200,000,000 binary digits. No type of office machinery which is now available could store or analyse such a mass of data fast enough or cheaply enough to satisfy the cost accountant of the firm. The digital computer could undertake the analysis if only the data could be transcribed and presented to it in a form it could accept.

Now that we have described a few typical commercial problems in some detail, we can discuss the conclusions that can be drawn from a year's study of the application of the Manchester machine to this type of work. Let us first remind ourselves that this machine, like most of the other digital computers in the world, was built for the use of the mathematicians and other scientific workers in a university, and that it was expected that they would use it to undertake research work of all kinds. We need not, therefore, be surprised if the machine in its present form is not ideally suited to work in a very different field.

Nevertheless there is no doubt that this machine and several others which now exist are likely to be of considerable use in commerce and that even now they are capable of handling many problems of great importance.

It is equally clear that the present machines will be even more useful as soon as more elaborate input-output mechanisms have been built for them. Data processing is an essential part of almost all commercial work. There is in fact a difference between ordinary business computations and the scientific work for which machines have been used in the past. It is a difference in degree rather than in kind which can be stated quite simply. Most scientific problems involve a large amount of calculation based on a relatively small amount of numerical data. Commercial computations, on the other hand, usually require only a few operations to be performed on each set of figures, so that a small amount of computation is needed, but it is usually necessary to process a great deal of data. We have already mentioned a computation which involved some hundreds of millions of digits but other problems are even more demanding. For example, the Metropolitan Assurance Company has in force about 30,000,000 policies, each of which, on the average, has to be referred to or altered about five times a year, and there are probably about 1,000 binary digits on each policy. The total number of digits which have to be processed per annum is of the order of 10^{11} which is about as many as there are stars in our galaxy. The 19,000 clerks whom the company employ are kept very busy.

It is quite clear that no machine will be able to take over the whole work of a big office in the foreseeable future. A large part of the work of any office consists of opening letters, reading and deciding how to deal with them, routing them to the proper departments, and coping with what always seems to be an unending series of special cases. No machine is likely to do this sort of thing as well as a competent clerk; not only can a clerk read almost anyone's handwriting, but he can easily make decisions which are, in fact, based on the experience of a lifetime. Even the simplest routine job may involve much more general "information" than can be retained in the memory of any existing machine, and moreover it may involve reasoning which, although it is well understood by the clerks themselves, would be very difficult to put down on paper in a form which the machine could understand.

There seems to be no chance at all that the machines will ever be able to usurp the primary functions of management and decide company policy in the broad sense. The "experience" of a computer is limited to the programmes which have been prepared for it and the data which it remembers. A machine might supply almost all the data on which decisions of policy must be based, but it could never handle the strategy of business—a machine shows up poorly when it has to ponder the strategy of a game of chess (see page 288). One can imagine that in a few years' time a single machine might replace hundreds of the clerks in an insurance office, but under no circumstances could it ever replace the board of directors.

Finally, and perhaps most important of all, these machines will offer to management the opportunity of completely revising current office procedure.

There are some computations which can be taken over in their present form and done by machines exactly as they are now done by clerks, but the machines will come into their own when office work has been redesigned to suit and to exploit the proposed machines. The organization of most offices has grown up piecemeal over a period of years, and the form which it takes has been dictated by the available machinery (using this word in the broadest possible sense). Many of the fundamental decisions which determined the form of the organization in the first place have long since been forgotten, but the whole structure of the office and its operations must now be reconsidered from first principles. This means, of course, that the planning will have to be done by senior executives, or by the "organization and methods branch" of the company concerned. These people alone will be able to take advantage of the opportunities which are offered by the new techniques and to decide what new methods should be used. They alone are likely to be able to resist the insidious temptation to place confidence in conclusions derived from unreliable data which have acquired a certain sanctity in many minds merely because of the complexity of the arithmetic operations which have been performed upon them.

We must analyse the implications of a most important point which has so far been glossed over but which must now be brought into the open; we must discuss the *programming* of commercial problems for digital machines. It always outrages pure mathematicians when they encounter commercial work for the first time to find how difficult it is. It is often much harder to programme a simple commercial problem for a digital computer than it is to handle quite sophisticated mathematics on the same machine. The reason is not far to seek; the behaviour of human beings and of their institutions is far more complicated, and, despite Heisenberg, far less predictable than that of elementary particles.* Most of the work of preparing a typical commercial programme consists in arranging to handle and to process an enormous mass of data, and to look after all possible cases and all the exceptions to every rule; these are questions of logic and expediency rather than of mathematics, and as such are hard to manage. A typical research problem will have been formulated for the mathematician by a physicist or an engineer. The underlying physical principles may not be understood by anyone, they may well deal with the fundamental mysteries of nature, but the mathematical formulation of the problem implies that the computation is reasonably straightforward. On the other hand a typical commercial computation is probably handled by several hundred clerks. Each individual operation is perfectly straightforward, and there is no mystery about the underlying principles, but because of the complexity and the ramifications of the work it may well be that no single individual understands the office procedure in detail, so that the would-be programmer may have to spend months in finding out what is in fact done before he can draw up his flow diagram. This sort of thing inevitably makes for complexity in programming. It is a tribute to the skill and the aesthetic sense of physicists and mathematicians that the equations which they have produced to describe natural phenomena

* Heisenberg showed that it is impossible for an observer ever to know precisely both the position and the velocity of an electron at the same time, and that in trying to measure one of these quantities the other is inevitably altered by an unknown amount. The impossibility of ever learning all the facts involved in modern business or of undertaking completely rational planning is discussed in Chapter 23, and it is worth pointing out that any part of a business which is subjected to a detailed operational analysis will never be quite the same again.

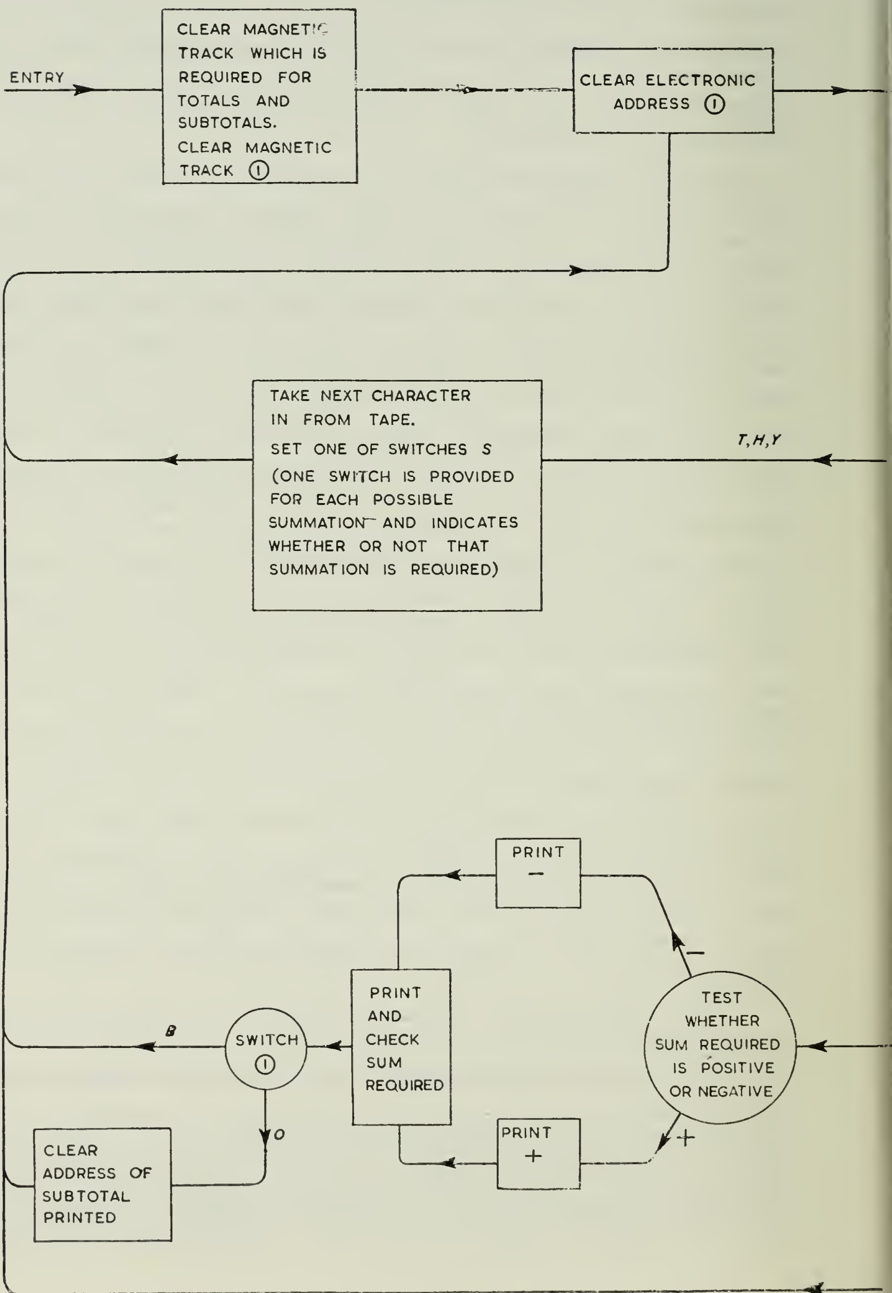
are relatively simple and straightforward. Sir James Jeans went so far as to assert that the Almighty must be a mathematician Himself, or otherwise natural phenomena could not be described in such beautiful mathematical terms. This argument takes us rather out of our depth, but we hope we shall not be accused of impiety if we remark that it is clear that no such feeling for beauty inspires the Board of Inland Revenue.

It will be clear therefore that the preparation of the programmes which will be necessary to enable digital computers to handle commercial work will be both difficult and tedious. On the one hand it is true that once a commercial programme is known to be correct it may be used every day for years, but on the other hand it is so hard in practice to get any programme right that several mathematicians may be needed to look after a big machine in an office. We must stress the fact, moreover, that these people must have the same familiarity with commercial work that mathematicians are expected to have with differential equations, and that there is room in the organization both for the skilled accountant, who will have to be responsible for the flow sheets, and for more ordinary (and less highly paid) individuals who interpret the flow sheets for the machine. Furthermore, our limited experience suggests that as soon as a programme is working satisfactorily and the mathematicians have begun to breathe freely again, some executive comes along with the suggestion: "If the routine were altered in such and such a way, it would then be possible to compute so and so." This means, of course, that the machine is likely to be properly exploited and that the management is taking an enlightened view of its potentialities, but it also means that the wretched mathematicians have to begin all over again. It implies, moreover, that it is most improbable that one firm will be able to take over many of the routines which have been produced by another. It seems, in fact, that experienced programmers will always be in demand.

The statistical problem shown in Chart III took three times as long to programme as the solution of a dynamical problem involving thirty-one simultaneous differential equations with a couple of non-linear terms thrown in for good measure. Any mathematician would regard this as quite a difficult calculation; but the routine for solving it on a computer is less than half as long as that required to do the statistical problem. A similar problem in sales analysis which was done in America took nine mathematician-months to

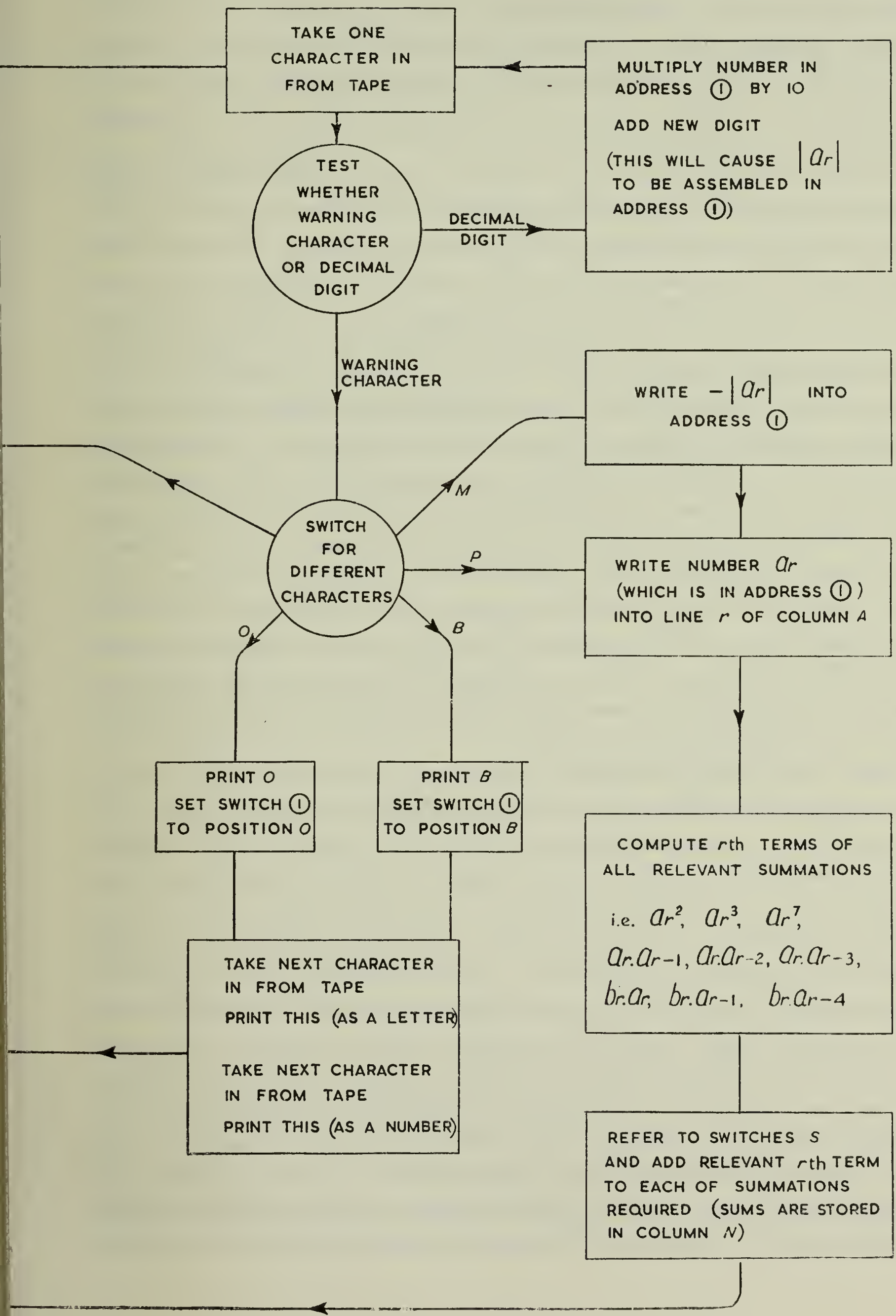
CHART III. FLOW SHEET FOR A PROBLEM
PART OF A MULTIPLE

This diagram shows the method of computation of sums



IN STATISTICS WHICH WAS DONE AS A
REGRESSION ANALYSIS

of various powers and products ($\Sigma a r^n$, $\Sigma a r a r{-}s$, $\Sigma b r a r{-}s$)



programme, and fifteen machine-hours to get rid of errors: it will be run for an hour and a half once every month.

Before we discuss the progress which is now being made in the construction of new attachments for existing machines, it is interesting to speculate for a moment on the possible long-term development of the complicated specialized equipment which will have to be designed to handle the work of the great offices, for example, the insurance companies[(1)] and the banks.[(2)] Machinery of this type will almost certainly have to be built to special order. The largest and most expensive piece of equipment that can now be bought from a catalogue is probably a lorry or a motor car. Almost everything larger is "custom built," so it is only to be expected that these machines, which will become an integral part of old-established and most complicated enterprises, will have to be designed after long and detailed study of all the operational requirements they have to meet, and after the most careful collaboration between the customer and the engineer. Much of the impetus for the development of such machines will have to come from the customers themselves, and it is remarkable that insurance and banking are almost the only large industries which have never set up research departments to sponsor the development of the machinery they need—and this in spite of the money they habitually spend on advertising. Even the film studios, which for so many years seemed to buy all their science from travelling salesmen, do better than this nowadays, and they have set up an organization to advise industry of their operational requirements.

There are in England about 5,000,000 bank accounts, and it has been estimated that for each one the banks invest approximately £5 in machinery; nevertheless the bankers have spent only minute sums on research into better and simpler methods of handling their book-keeping. It will be necessary to produce robot cheques as well as automatic accounting systems. A survey which was recently made of the possible application of such methods in a big bank suggested that savings of up to 80 per cent might be possible; in other words it might cost only a penny or two to process a cheque, instead of the shilling which it costs today. All United States government cheques are drawn on punched cards in order to facilitate subsequent sorting. The more conservative English banks have rejected this idea on the grounds that many cheques are damaged by being carried around in people's pockets. It is probable, however, that the majority of cheques, particularly those concerned with transactions between business houses, could be preserved intact so that they would pass through the sorting machines; the few damaged cheques could then be sorted by hand.

We must now inquire what improvements can be made in the next few years to our standard digital computers, and try to assess how they can be fitted to take over the commercial work which it seems clear they are destined to do.

Input Mechanisms

It will be necessary to develop methods for transferring the information on written documents into the machine in the cheapest and most expeditious manner. The technique of transferring data to punch cards has been intensively studied for many years, and the process is efficient and relatively cheap. Since it is easy for the machine to absorb information from cards at high speed, there is no reason why cards should not be used, at least in the early stages of development, as an intermediary between the original documents and the machine. This process would incidentally have the advantage that all the techniques which already exist for sorting and handling punch cards could be taken over *in toto*. Furthermore one must not forget that a very large amount of capital has been locked up in the form of data punched on cards by many firms during the past twenty years. Nevertheless all transcription processes are uneconomic and should be avoided if possible. The cards themselves are not cheap, and if they are used in large numbers, their storage is a problem. It would be far better if the information in the original documents could be fed directly into the machine. There are already methods by which pencil marks on a card can be read photoelectrically and made to operate a punch, so if the position of a mark on a piece of paper can be made to define its significance, the information so written could be fed directly into a computer. We have explained elsewhere why it would be difficult to make a machine which will read figures written in a variety of different handwritings, or to understand numbers which are dictated to it. The much simpler system which we suggest could be used, for example, to process bills produced in restaurants by waitresses, to handle a list of invoices from a standard list, or to make an inventory of items in a large store. As, however, numerical accuracy would be dependent upon the precision with which pencil marks were made, schemes of this kind are most suited to systems in which a statistically small error is unimportant, or in which errors are self-correcting, for example in the recording of readings from gas or electricity meters, where an error made in one reading will be corrected at the next. Some automatic printing processes can obviously be adapted to punch cards so that the computer can accept them. For example, the times at which operatives enter and leave a factory are automatically recorded on clock cards. The clock could be made to punch the cards instead of stamping them. Furthermore, if information is to be typed at some stage, the typewriter could be made to produce both the normal typescript and at the same time some form of record such as a punched paper tape which the machine can read, and at a later stage, perhaps it may become possible to eliminate the typescript entirely.

To summarize; every attempt will have to be made to feed original documents into the machine wherever this is possible, otherwise it will be necessary to use the simplest possible method of transcription, and until more economical methods are available there may be much to be said for the use of punched cards of some sort. Ultimately it may be practicable to record directly on to magnetic tape, which the machine can read even more easily than cards. If the machine has to be used to analyse and reduce experimental data every attempt should be made to convert meter readings directly into digital form in order to feed them into the machine automatically.

A digital computer which had a punch-card input and output mechanism would be able to undertake much of the work which is now done by card-processing machines, and do it far more quickly than it can be done today. This fact in itself may be sufficient to recommend such a machine to statisticians. A card reader could read the information contained in (say) 30 columns of a card, and feed the whole of this data into the machine, which could perform perhaps 500 numerical operations on it during the half-second or so which it takes to pass a card through a standard machine. Statisticians and others who concern themselves with the organization and classification of data would find it most useful, for it would quickly make available to them summaries which take a long time to prepare by existing methods. This is a technique which would be used in the analysis of costs which we mentioned earlier in the chapter.

Suppose we are classifying data from cards in a table containing 200 columns and 100 lines, i.e. 20,000 "cells." In order to prepare this table by conventional methods it would be necessary to sort the cards first by columns and then by lines. A machine which had a memory comparable to that of the Manchester University machine (32,000 six digit numbers) could sort the data into 20,000 cells in one passage of the cards, and simultaneously could store the results of several hundred arithmetical operations on the data on each card in 12,000 other cells. Alternatively, of course, the machine could prepare several smaller tables at the same time, during a single passage of the cards. The speed with which the machine would accept the cards would be limited by the number of cells to whose content any particular card contributed, and it might be reduced if the information on each card had to be added to the contents of more than a dozen cells in the table.

This technique has obvious applications, for example to sales analysis; the ways in which it could assist in the analysis of the census of population and the census of production are discussed in Chapter 20 so we shall not elaborate the point further here but merely note that this development may be an important by-product of the current developments of digital computers.

Output Mechanisms

Few scientific computations require a very high speed of output printing, and most of the machines which have been built so far have made use of standard teleprinter typewriters. These are inherently slow because they print only one character at a time, though even an automatic typewriter can produce so many figures in a working day that one sometimes wonders what can be done with them except to feed them into another machine. For commercial work, on the other hand, a great deal of output is necessary, and the Manchester computer can be made to feed a parallel printer of the type which has been made for the "tabulators" or punch-card machines. It will print 150 rows of 64 digits in a minute, which is more than 20 times as fast as a teleprinter typewriter; and this machine prints out the pay slips ior 3,500 people in an hour. Since the computer continues to calculate during the operation of the printer, it seems that in this computation at least, a printer of this type is fast enough to cope with the output of the machine, which we recall would take about an hour to work out the answers (though considerably longer to absorb the necessary input data). Alternatively there is a teleprinter-like punch which will punch 50 characters a second on tape. The tape then feeds half a dozen ordinary electric typewriters or a more elaborate machine like a "Flexowriter" or a "Varityper." It could even operate a Monotype machine which would automatically set up the results in type, thereby fulfilling the ambitions and ideas which Babbage propounded about a hundred years ago. This process is even quicker if magnetic tape is used intermediately, as is done on several machines in the U.S.A.

It would be possible to devise very much faster output printers, based, for example, on the direct photography of a cathode-ray-tube face, on to which the numbers are written by a television-like raster. At this stage, however, it seems improbable that such exceedingly fast systems will be needed. The binary-decimal conversion takes some time to do, and if one wished to print all 64 digits in every row, the speed of the Manchester University machine or of the E.D.S.A.C. is such that only about 100 rows of figures could be computed every minute. It follows that these new printers would handle the output of a computer which was considerably faster than either of these machines. In most normal calculations the time required for binary-decimal conversion is only a fraction of the total computation time.

Conventional printers are likely to be able to handle the output of the computer in most cases, and as a rule it is desirable to restrict the amount of printed material as much as possible; once it has been printed someone has to read it! It will be much better for the machine to retain within itself all the data which it may require for its own use, and which need not be inspected by an accountant, and to print out only those figures which actually have to be seen.

We have not, so far, considered the problems involved in making the machine write letters to the Income Tax Inspectors. Even this process might be mechanized to some extent; the machine is already able to indicate some of the occasions on which a letter must be written. This is legally required for example, if, on his first payday, more than five pounds excess income tax has to be refunded to a workman, and it is only a step from deciding that a letter must be written to writing it out in such cases as this. The whole process of correspondence with the taxation authorities will probably only be mechanized when they too use a machine of the same type! Even then it may be necessary to have a few clerks to cope with the unusual and unexpected cases which occur too infrequently to justify the effort of programming the machine to handle them.

Memory Circuits

Machines which have been built in this country so far have had two types of memory, one of limited capacity, holding perhaps

10,000 binary digits any of which can be recalled in a millisecond or less, and a larger memory, usually on a magnetic drum, which can hold several hundred thousand digits, but which takes about thirty milliseconds to extract any particular number. Both types are relatively expensive and bulky, and it would be uneconomic to increase the capacity of either by a large factor, though one could imagine a high-speed store containing 50,000 binary digits, or a drum holding two or three million. Now a magnetic drum could hold all the instructions which a machine would require in ordinary commercial computations, but our study of the problem has shown that its capacity would be much too small for some of the jobs which one would like to do. It would, for example, be uneconomical to use a drum merely to store the data which would be needed from week to week to work out the wages in a small factory, and the capacity of the drum would be far too small to handle the analysis of costing in the same factory.

We have discussed in another chapter the possibility of making a memory of very large capacity on magnetic tape, of the type which is now used for sound recording, and we have shown that it would be quite possible to make a store which would hold several hundred million binary digits. Such a store could remember all the contents of an encyclopedia. All the account books of a firm could therefore be kept in a medium which is admirably suited to a digital computer, which could read from it and write into it easily and at great speed.

The problem of sorting, classifying and analysing such a vast amount of data has yet to be solved. It will be necessary to make certain that the machine can obtain rapid access to any part of the stored information which it may need. Probably the magnetic disc equivalent of a "juke box" will have to be developed to do this; or perhaps a magnetic-wire record could be wound on to a large number of spools, so that blocks of data could be transferred from the tape to the drum before analysis. Moreover it will be necessary to impose restrictions on the order in which the calculations are performed.

If the input data were sorted before being fed into the machine they could be combined with data already on the drum without difficulty. This sorting can, under certain circumstances, be rather laborious, and much of it could be avoided if the information on the tape were transferred in blocks of perhaps half a million digits on to a magnetic drum, an operation which need take only a few minutes per "block." It would then be necessary to sort the input data into relatively large blocks each corresponding to a drum-full of recorded data. This is true because any item of information on the drum is available within thirty milliseconds, so that the machine itself could "marry" information coming from any card in the batch with the appropriate stored information in the drum. A crude sorting such as would be needed would be very easy and quick and of course in many cases the data is roughly sorted automatically as it is prepared, so that it is probable that the problems associated with the sorting and handling of tape will be easier to solve than those involved in handling ordinary papers.

It is important to emphasize the fact that any part of the data which had been stored on magnetic tape could be printed out if it were needed, so that doubtful points could be studied by an accountant. This point may be important from a purely legal standpoint. It is only a few years since the Income Tax authorities first accepted accounts which were kept solely on punch cards. Until then many firms kept what were to all intents and purposes two sets of books, one on punch cards, and the other a direct transcription on paper. For the past five years or so it has been permissible to use punch cards alone, if any sections of the accounts which are chosen by the Inspector can be printed out for him to examine. A digital computer could operate in the same way. This particular difficulty would rarely arise in statistical work.

The storage of such items as the addresses of customers presents still other problems. They could be stored on tape, but if so they would only be accessible in the order in which they had been recorded. It will probably be necessary to develop a store which will hold a few hundred thousand "addresses," any one of which can be read in a fraction of a second. The development of such a store is proceeding.

General Considerations

The machines which we have considered so far have all been large and expensive—a fact which suggests that at this stage only a few of the biggest firms will be able to afford to install machines solely for accountancy. Many firms engaged in engineering and scientific development, optical work, crystallography, and so on, may find it possible to use a machine part-time on their professional computations, and part-time for accountancy. Such an arrangement would be most useful, and would exploit all the potentialities of the machine.

We must not exaggerate the cost of these machines, however. No machine which we can now foresee should cost much more than a hundred thousand pounds to install, complete with an elaborate input-output mechanism and a large memory, and it should not cost more than a few thousand pounds a year to run. If the machine can replace three or four hundred clerks, each of whom may cost the firm a total of six hundred pounds a year, the machine might save more than its capital cost in one year, and yield a handsome profit for the rest of its life.

There can have been few more promising investments since the Duke of Bridgewater built his famous canal a hundred and ninety years ago, at the beginning of the first Industrial Revolution. It cost £250,000; it enabled him to halve the cost of transporting coal from the pit head to Manchester, and was finally sold for £750,000. Soon afterwards (in 1812) the American Government allied itself with Napoleon Bonaparte and declared war on England; it tried to float a loan to pay for the war but could not borrow as much as many contemporary Englishmen were able to raise on their private credit and invest in the new factories which the Industrial Revolution demanded.

Smaller, simpler, and cheaper machines will undoubtedly be built within the next few years; they will be suitable for installation in smaller offices and will do much the same type of work as we have been discussing.

In much the same way as the modern system of mass production has supplemented the craftsman but has not been able to replace him, these machines will supplement the work of the accountant; they will give him a far greater control over his medium, and will if anything increase his importance in the scheme of things, for not only will he have to set up and programme the machine, but he will have to interpret its results, which will be far more detailed than anything which he has at present.

Moreover he will be able to use the results of his analysis much more effectively because they will be available to him when they are still fresh enough to be of real use. Every electrical engineer knows how difficult it is to control a servo system if the "error" signals have been delayed too long before they are used. The whole system may easily get out of control, oscillate and "hunt" about its equilibrium position, and every attempt to tighten up the control makes matters worse. There may well be a lesson here for all would-be planners, for the analogy between the running of a business and the control of a servo system, though it may be unfamiliar to business men, is quite a good one. A servo system may be roughly defined as a mechanism which is controlled at all times by the difference between its position (or its velocity) and the position (or velocity) which it ought to have. For example the motors which turn a big gun which is following a target might be worked by a man who controlled them so that he attempted always to keep the angle between the actual position of the gun, and the position in which it should be, as small as possible. The changes he would make to the motor controls would be determined by the error in the position of the gun. It is of course possible to devise a system which does not require the intervention of a human being—for example an automatic helmsman in a ship and an automatic pilot are typical servo systems; the behaviour of a man driving a car is very similar to that of these automatic devices, and can usually be described at least approximately by the same equations. If for any reason, the error signal is not received for some time, the operator will tend to over-correct, and will pass through the proper position without realizing it—he will then correct in the opposite direction, and unless he is careful the whole system may oscillate violently about its equilibrium position. The more vigorously he operates the controls the more violent the oscillations become. Now the mathematics of servo systems has been intensively studied, and it is well known that if the error data are delayed before they are used, the servo can be stabilized only by using information concerning both the error itself and its rate of change with time, and that the determination of rate of change is possible only if the error data (though stale) are known with precision.

The delays which are to be expected at the present time in preparing books of account, and the difficulty involved in making reliable estimates of costs or analyses of sales, mean that executives have to try to determine their policy from information which is both stale and unreliable. They may know what they would like to do, but they never know (until it is too late) what it is they have in fact done, nor do they know accurately how much better they are this month than they were last. No wonder the art of "steering" a business is so hard to learn, and that it is next to impossible to forecast business trends with confidence. The use of digital computers should improve the reliability and the precision of the data which an executive needs, and give them to him while they are still fresh, so that he can use them with confidence to guide the operation of the business for which he is responsible.

The introduction of digital computers may have an effect on clerical work comparable with the effect on manual work of the introduction of machines in the Industrial Revolution. It is well known that modern machinery has increased both the number of people employed and the output per man, and in much the same way we may hope that not only will clerical machinery be used to relieve human beings of the tedium of the routine jobs to which many of them now devote their lives, but also, by allowing the cheap and rapid computation of data which are now inaccessible to us, the machines will allow us to improve our understanding and control of the complex organizations which produce and distribute the goods upon which our material civilization depends.

REFERENCES

1. *Report of the Committee on New Recording Means and Computing Devices.* Society of Actuaries of North America (Sept., 1952).
2. Davies, A. E. "Banking by Electronics," *The Journal of the Institute of Bankers* (Oct., 1952), 265.

Chapter 23

ELECTRONIC MACHINES AND ECONOMICS

Somehow it seems to fill my head with ideas—only I don't know exactly what they are!—Lewis Carroll

Not only are electronic machines likely to have a profound influence on the thinking and working of economists, but economic considerations must determine the uses to which these machines are put. While the latter point need not be pursued at length here, it is clear that some of the questions raised by this latest industrial revolution are primarily for the economist to discuss. In the narrower sense he will have to consider whether the construction and use of such machines is "economical," whether and to what extent it will pay to substitute them for human labour; in the wider context the emergence of new methods of automatic control may give rise to problems of economic and social adaptation to a rapidly changing technological structure. Again, it is for the economist, drawing on past experience and his awareness of the interdependence of technical and social phenomena, to help in making this revolution less painful than earlier ones have been and to suggest how to make the best use of it. In our world we cannot afford to ignore any of these aspects. For the present, however, the main emphasis will be on the application of electronic machines to economic problems.

Some Examples

The first application of high-speed computers (of the analogue variety) to economics was made by W. W. Leontief in his study, *The Structure of American Economy*, 1919–1929.[(1)] The conception of this work was fairly simple* and not entirely original. Some 180 years earlier, Quesnay had published his *Tableau Economique*; both economists attempted to describe the structure of a national economy by following the movements of goods and services from one field of activity to others.

* The distinction between what is simple and what is complicated is rather arbitrary. While all parts may in themselves be simple, there may be so many of them and their inter-relations may be so numerous as to make the whole very complicated indeed. Recent developments of economic analysis tend towards "simplicity" of this type.

One industry, e.g. the iron and steel industry, may deliver goods to individuals for direct consumption, or to other industries for further processing or use—rails to the railways, sheet metal to the motor industry, girders to structural engineers, etc. If we have sufficient information we can show how the output of each industry is distributed among other industries and households. Similarly, each industry needs goods and services from other industries; to produce iron one needs coal, iron ore, human labour, machinery, etc. Again, it may be possible to show where the raw materials for each industry come from. If all this information is available one can construct so-called *input-output tables* showing both the sources from which each industry obtains its goods and the destination of its products. Very often it happens that there is two-way dependence between industries; e.g. the iron and steel industry may supply goods to Transport and obtain services from Transport. In modern communities the degree of interdependence among industries may be very high.

Quite clearly, we cannot trace every single commodity; there may be several million different products produced and consumed. If we are to collect our information in broader groupings, for instance by some accepted industrial classification, our input-output tables will have to show values of groups rather than of individual commodities. Once we have these tables we are, in principle at least, able to answer questions such as—

"If there is a primary increase in the price of coal, what will be the ultimate effect on the price of steel, allowing for the fact that a higher price of coal will also raise the cost of transport and of other industries contributing directly or indirectly to the cost of steel; and also taking account of the fact that an increase in the price of steel will raise the cost of coal still further?" Or: "If there is less electricity available, what will be the effect on all the other industries and on the economy as a whole?"

If such questions are to be answered within the present framework, one must have the proviso that the *structure* as represented by the input-output table remains unchanged.*

Even if, for simplicity, or for lack of adequate information, we tabulate our data under a small number of industrial headings—and fewer than 30 groups would give a most inadequate picture of

* In this argument we assume that the consumption of coal by the steel industry is unaffected by a change in price. Leontief analysis in general is subject to the same restrictive assumptions.

the economy as a whole—the work involved in answering such questions without using efficient computing devices would be very lengthy indeed. To be of any real use the table would have to include particulars of a much larger number of industries. Work along these lines is in fact proceeding in several countries. Nobody but an economic historian would expend much time and energy obtaining information if the answers laboriously deduced from it were of doubtful value or if the computation would take so long that the answers it produced were no longer of any practical interest. The use of punch-card methods and of electronic machines makes this kind of analysis of practical day-to-day use.

As another example let us consider the case of a firm with several workshops, each of which has a weekly capacity of a certain number of man hours. The firm has a number of jobs offered to it, each of which it can accept or reject. Each job requires so many hours work in each of the various workshops and brings in a certain profit. If the firm cannot take on all the possible jobs because of the limitation of its capacity it will have to accept some jobs and reject others. Since each job brings a certain amount of profit, the firm will try to make such a selection of jobs that its profits each week are as high as possible, while keeping at the same time within the capacity of each shop. If it is possible to accept some jobs only partially—say only one half—leaving the other half over for the next week or perhaps subcontracting part of it, the whole problem can be set up in a table with as many columns as there are jobs and as many rows as there are shops plus jobs taken together. For example, if there are five shops and twenty jobs offered, the table will have 25 rows and 20 columns. The jobs which should be accepted can then be found by systematic trial-and-error in the form of a two-person "game."* In the worst case one might have to solve twenty linear equations in twenty unknowns twice over, which is a formidable undertaking.†

With such a number of rows and columns the solution may be just within the capacity of ordinary calculating machines, or at any rate of standard punch-card equipment. A sufficiently speedy answer in any realistic case, however, requires the use of electronic

* See page 277.

† It must be realized that even in this form the problem is somewhat artificial in that it leaves out a number of complications such as the possibility of working overtime and the adjustment of prices which are charged by the firm. These factors could be allowed for at the expense of a considerable increase in the number of rows and columns.

equipment. Whether it will in fact be worth finding the exact answer in this fashion depends on two things—

(*a*) The possible financial loss which results from using rule-of-thumb decisions as to which jobs to accept (this loss will be small if the firm has sufficient accumulated experience and if the collection of jobs offered does not vary greatly from week to week).

(*b*) The cost of carrying out the computations. If the price of hiring electronic machines is sufficiently low, it will probably pay to obtain the exact answer at least occasionally as a check on the adequacy of rule-of-thumb decisions.

One further example may be quoted. If we know the prices of all available foods, the nutrient contents of each food (i.e. the amount of calories, protein, vitamins, minerals, etc., contained in one pound of bread, meat, milk, potatoes, etc.), we may want to find the cheapest combination of goods which satisfies specified dietary requirements.* This problem was once tackled for 77 foods and 9 essential nutrients when the solution required 120 man-days using ordinary desk equipment.(2) This is much too long for most practical purposes. However, better numerical methods have been devised and these, coupled with the use of electronic machines, reduce the time for solution considerably.

Meteorology and Engineering in Economics

There are two basic aims of economic research; one comprises the description of how people behave and how they can be expected to behave in the future—this may be compared to the descriptive and forecasting part of meteorology. The other aim is to decide on the best way of doing something if one already knows how people react, and this may be compared to the aim of the engineer. The last two examples given above are typical of the latter class. In the food problem we are interested in how to reach a given goal most "economically" by making a suitable choice of the means at our disposal. In the problem of workshops and the jobs among which a choice has to be made we are trying to find the best possible use of limited capacity. Economic problems of the engineering type can generally be expressed as instructions in one or other of these forms: "Reach a given target most efficiently," or: "Make the most of limited

* It is possible to allow for further details such as the condition that at least one-half of all protein should be obtained from foods of animal origin; this increases the amount of computing work. In order to plan a realistic low-cost diet allowing for such things as personal taste and food habits one would have to consider many more conditions than the dozen or so food factors for which the requirements are accurately known.

resources." The problem which was first mentioned, namely to estimate the effect on the economy of an increase in the price of coal, is an example of "meteorological forecasting." We want to know what will happen if certain changes are introduced into our system and people continue to behave as they did.

In practice the distinction between the two types of problem is never clear cut. To the extent that the changes in some factors are due to policy decisions, and not to external physical or psychological causes outside one's control, one can attempt to evaluate the consequences of any particular line of action, and choose the one that seems the "best." In particular, a modified Leontief analysis has been used to forecast the effect on employment of the expansion of certain industries, allowing for all secondary increases in other industries. If there were unemployment the authorities should try to stimulate expansion in that industry which yielded the greatest increase in employment all round. This technique can also be used in dealing with questions of regional development.

In the United States the Bureau of Labour Statistics has been working out such "full-employment patterns," and of the many extensions of the Leontief technique this is one of the most important and topical.

It is obvious that not only industries in any one country, but also countries themselves, are interdependent. A change in the situation of one country, for example a poor harvest, may affect employment in the export industries of another, the balance of payments of a third, and so on. Most of the time, however, economists have to focus their attention on a very small selection from all the factors which significantly influence economic events and the best they can do is to be aware of the limited validity of their conclusions.

In recent years a new technique for dealing with the "engineering" side of economic problems has come into being. This technique which can be thought of as an extension of Leontief's method is called linear programming.(3) It does not attempt to investigate the behaviour pattern of people so much as the physical conditions under which individuals, firms or nations operate, and it seeks to find the best way of reaching given aims or getting the most from limited resources. The economic structure can be described either by a large number of simple (linear) relationships, or by a relatively small number of more complicated statements, i.e. by a large number of straight lines, or by the continuous curve which they represent with a sufficient degree of accuracy for the purpose in hand. The linear programming technique depends on the use of a large number of simple relationships between the economic units concerned.

Because of the cost involved in collecting and analysing so much information, the economist must necessarily restrict his outlook; electronic machines will relieve this situation by making analysis cheaper. However, some of these machines suffer from comparable shortcomings because of their restricted storage capacity.

The cost of the analysis can be divided into two parts. First, we have the capital cost of the machine which will increase with its complexity and with the speed at which it is capable of operating. Secondly, we have the running cost of the machine itself and of the associated organization which it becomes necessary to build up in order to ensure that up-to-date information can be presented to the machine. There is another very important factor which we must not overlook; information becomes of little value if it is not available at the right time. Thus, an economist or manager may find it cheaper and better to act on imperfect information if the collection and digestion of data is costly and takes too long. From this it can be seen that rationality in planning cannot be defined precisely. As mentioned in the example above, perfect "rationality" in the solution of a given problem might be too costly (even where possible) and hence it would be irrational.[(4)]

To the extent, however, that collection of information and computation are "cheap" relative to the cost of a wrong judgment—which seems probable for many economic activities—the linear programming technique and the employment of high-speed computers are undoubtedly useful and can be expected to increase in the future.

The Theory of Games

When J. von Neumann and O. Morgenstern published *The Theory of Games and Economic Behavior* in 1944, economics took a long stride forward. For electronic machines provide simplified models of some of the functions of the brain, while "games" are simplified models of economic behaviour. The inter-relation between man and machine is strengthened by the fact that machines can be made to play games and that economic life can be thought of as "solving" highly complicated computational problems.

Let us then look at the theory of games. Many of the factors which define games are similar to those occurring in real life; in both, there are "rules" of conduct which it is either impossible or expensive to break. In our society one must not steal; in chess the moves of the pieces are clearly defined; in bridge one has to pay a penalty, a high one, for a revoke.* As long as nobody questions the rules the game can proceed. A revolution, or new legislation, changes the rules, and alters the character of the game or the society.

Secondly chance plays a part in both games and real life. There are, of course, games (like chess) where ideally there is no element of chance involved, but for a large number of games chance has its say. The intention of shuffling cards is to give equal chances to all the players.† Anything which cannot be predicted precisely, for example the weather, is the counterpart of luck or chance in games.

The most important features of the theory of games, however, are the determination of a "good" strategy, and the emergence of "coalitions." Corresponding to coalitions in games we have monopolies, unions and the like in economics, and the investigation of the effects and the stability of such forms of organization is undoubtedly of importance.

Neither in games nor in economic life need perfect competition be regarded as logically necessary or practically permanent since it implies the absence of coalitions among players. One of the reasons for the impermanence of competition may be the cost of information in the widest sense. Economists were in the habit of assuming immediate and complete knowledge of all the relevant facts by all participants in the great game of making a living. This would imply the transmission of all the relevant information and its complete analysis at no cost to the users, which is clearly impossible.

The question of what constitutes a "good" strategy for an individual player is answered in the following example of a game played by two people. The loser pays a certain amount to the winner. Either player may have several lines of action, called strategies, and corresponding to each strategy a player follows, he wins or loses a certain amount. Whatever the second player does the first player can make sure of winning no less or losing no more than a certain amount if he selects that strategy which is best for him.

* Moreover the cost of obtaining information plays a significant part in the bidding in bridge. Most bids which supply information raise the level of the contract and with it the possible bonus or penalty. This is the real justification of bidding "conventions" such as the Blackwood slam convention.

† The impetus given to the development of the theory of probability by gamblers is well known. The Theory of Games is not concerned with this, but uses it as one of its foundations.

A numerical example may make this clearer. In the table below, the players—let us call them Jones and Robinson—have two strategies each. The strategies available to Jones are represented by rows, those to Robinson by columns—

JONES	ROBINSON	
	Strategy I	Strategy II
Strategy I .	Pence 3	Pence 2
Strategy II .	4	– 1

The square on the upper left shows how many pence Jones will get from Robinson if Jones selects his strategy I and Robinson his strategy I. Under these conditions the table shows that Jones wins 3d. from Robinson. The next entry 2d. corresponds to Strategy I of Jones and Strategy II of Robinson. The lower right shows that if Jones chooses his second strategy and Robinson his second strategy, Jones will lose one penny. Now if Jones chooses his first strategy he cannot win less than 2d, whatever Robinson does. Similarly, Robinson by choosing his second strategy cannot lose more than 2d. whatever Jones does. In this simple case, Jones will protect himself by playing his first strategy and Robinson by playing his second strategy. No matter what his opponent does, neither player has any incentive to change. Even in this simple case, however, there may be complications where the players find their best way of playing by combining the strategies available to them in certain proportions. If both players have more than two strategies at their disposal, the solution of the problem is found by obtaining the best combination of many strategies, a process which may be rather laborious.

A further and more complicated example is given in the article on Chess, see page 288. In this game the possible number of strategies is very large and even the numerical assessment of the value of each strategy is very difficult.

The importance of this sort of game lies in the fact that certain economic problems, such as the second and third examples given above, can be formulated as two-person games.* In the problem of

* This type of problem is that of minimizing or maximizing a linear function subject to linear constraints, and can be shown in certain cases to be formally equivalent to a game. The solution involves finding a square matrix in which the row and column sums of co-factors are non-negative.

finding the cheapest combination of foods, for instance, we can think of "Man" playing a game against an imaginary opponent. "Man" will, by finding his best strategy combination, try to win as much as his opponent, imposing conditions as to the nutrients he requires, will let him. In this case perfect information can be assumed to be available to both players and the opponent is assumed to play as well as can be.

This translation of simple economic problems into a game can be very helpful, both for logical clarity and for efficient numerical solution. If the number of strategies is at all large computation becomes too long and costly without high-speed computers.

Conclusion

It seems probable that many arguments about the respective merits of planning and the free play of the price mechanism are meaningless. The issue which must be decided in any particular case is that of overall cost, including that of obtaining relevant information and the administrative cost of acting on it. It may be cheaper to use the price mechanism to determine the use of resources in certain fields of activity, and to use commodity balance sheets, input-output tables and linear programming in other cases. It is possible to use the answers obtained by means of the linear programming technique of physical planning as a framework within which the price mechanism is made to do the work of allocating scarce resources and of deciding between possible alternatives. The "rationality" of a given planning procedure may not be a precise concept but it is likely that in a large number of cases numerical evaluation will be cheaper than target guessing even when the data are subject to considerable error.

Bad planning can be worse than no planning at all since this implies that one ignores too many of the relations that exist simultaneously in any economic situation, and that one fails to take account of the large errors which exist in most economic data. On the other hand, one should not carry the belief in pre-established harmony too far. One must not exaggerate the economy of leaving as much as possible to the "hidden hand."

Economic life can be a very inefficient computing machine, not merely in terms of running cost, but because of the frequent lack of stability of the solution. Alternations of boom and slump are a case in point. The direct advantage gained from the use of modern machines is due to their high computational speed and to their memory which makes it possible to allow both for the existence of the many cross-relationships of real life and also for the approximations in our assumptions and data, qualifications which one often ignores. Most important of all—if the speed of the machines is adequate it is possible for them to predict the results of a given course of conduct more quickly than the economic machine could reveal them. It would then be easier to choose amongst a variety of alternative policies that one which would be of most benefit to the economy of the country as a whole.

REFERENCES

1. Leontief, W. W. *Structure of American Economy* 1919–29. Harvard (1941).
2. Stigler, G. J. *J. of Farm Economics* **27** (May, 1945) pp. 303–14.
3. Koopmans, T. C. *Activity Analysis of Production and Allocation.* (New York: London, 1951).
4. Devons, E. *Planning in Practice.* Cambridge (1950).

Chapter 24

PROBLEMS OF DYNAMICAL ASTRONOMY

Some day there will arise a man who will demonstrate in what region of the heavens the comets take their way, why they travel so far from other planets, what their size, their nature—SENECA (3 B.C.–A.D. 65)

IN 1823, THE ROYAL ASTRONOMICAL SOCIETY awarded its gold medal to Babbage for his invention of the difference engine. In the course of his oration the President said:

"In no department of science or of the arts does this discovery promise to be so eminently useful as in Astronomy and its kindred sciences. In none are preparatory facilities more needful, in none is error more detrimental. The practical astronomer is interrupted in his pursuit, and is diverted from his task of observation by the irksome labour of computation; or his diligence in observing becomes ineffectual for want of yet greater industry of computation. Let the aid which computed tables afford be furnished to the utmost extent which mechanism has made possible through Mr. Babbage's invention, the most irksome portion of the astronomer's task is alleviated, and a fresh impulse is given to astronomical research."

The problems of dynamical astronomy have for many years occupied the attention of mathematicians. These problems may be roughly divided into two types; those of a theoretical nature where an algebraic approach is required, and those practical problems where numerical results are the ultimate objective. We shall not concern ourselves here with the first group but will concentrate entirely on the second. This includes such problems as the computation of orbits and search ephemerides from sets of observations of minor planets or comets and we shall consider this particular problem in greater detail. Such computations are at present generally done by hand but, given the necessary facilities for the use of an electronic computer, a great saving of time could be achieved.

The problem can be considered to consist of various stages. First, from a set of observations of the body, a preliminary orbit is computed. Using this orbit a search ephemeris is made, i.e. the positions of the body in space are predicted for times in the near future and, guided by these predictions, further observations are made from which a better orbit can be computed. The observations are given in terms of right ascension (R.A.) and declination (dec.) at given times, i.e. in terms of two angles which, together with the time, define the direction in space of the line between the observer and the body. Since the distance along the line is not known, the position in space is not fully determined. It is, however, known that, apart from perturbations arising from the gravitational fields of other planets, a planet will move in an elliptical path about the sun. To define this path, six elements are necessary and these can be determined from the six values of R.A. and dec. obtained by three observations. The equations connecting the elements with the observations are extremely complicated and cannot be solved directly. The usual process is to obtain the distance between the earth and the planet by a process of successive approximations (one well suited to automatic computation) and, having thus found the positions relative to the earth, those relative to the sun are obtained and the elements of the orbit derived from these.

Although, as a result of perturbations by other planets, the actual orbit is not strictly elliptical, this preliminary orbit can be used to predict the positions of the body for some time ahead. From the elements of the orbit, equations can be derived to give the R.A. and dec. at subsequent times. These predicted positions constitute the search ephemeris and are used as a guide to further observation. It may happen that, soon after a minor planet is discovered, it reaches a position where it is lost in the rays of the sun and, in this case, these predicted positions may be used to identify it again when it emerges. An improved orbit may be obtained by applying to the preliminary orbit corrections based on the subsequent observations. When the problem of predicting positions on future returns of the planet arises, the above-mentioned perturbations must be taken into account since, if the planet passes close to one of the larger planets such as Jupiter or Saturn, its path may deviate considerably from an ellipse. This problem is actually dealt with by a process of step-by-step integration which, although extremely tedious to do by hand, is easily adapted to machine computation.

When computing the preliminary orbit of a comet by hand, it is usual to assume that the orbit is parabolic rather than elliptic since this introduces certain simplifications into the computation. Comets are in general only visible in the portion of their path near the sun, and here the difference between the parabolic and elliptic orbits is so small that the accuracy of the observations is probably not sufficient to distinguish between them. Since a parabolic orbit is defined by five elements, only five of the six observed quantities need be used and the sixth can be used to test the "goodness of fit" of the parabola. If the agreement is not good it may be necessary to proceed to an elliptic orbit; also, if the comet is periodic, an elliptic orbit will have to be computed for predicting its return. Using a high-speed computer, a programme could probably be produced to find a general orbit, i.e. elliptic, parabolic or even hyperbolic according to the size of one of the elements, and this same programme could then be applied equally to minor planets or comets.

An average of five or six comets are discovered in a year, some of which may be identified with previously observed comets by methods similar to those described above. Of the minor planets, over a thousand have been observed well enough for reliable orbits to be computed, while several hundred others have been observed and lost again because the observations obtained were insufficient for the computation of accurate orbits.

Two major difficulties arise in the application of large-scale computers to this type of work. The first is the amount of reference which has to be made to the *Nautical Almanac*. The amount of initial data for determining the orbit is small—probably three observations of R.A., dec. and time, but for each observation about six references have to be made to the *Almanac*. Again, in calculating the ephemeris, data have to be extracted for each position calculated, and when dealing with perturbations, the co-ordinates of the major planets are needed over a period of several years. With most tabular functions it is fastest to store a subroutine for generating the function within the machine and find the values directly as required, but unfortunately the tables in the *Nautical Almanac* do not lend themselves to this treatment. However, high-order interpolation can be done at great speed and, given sufficient storage space, it should be possible to store some of the tables at wide intervals and interpolate in them. This applies particularly to the co-ordinates of the major planets and these could be used for all perturbations over a period of years, but it is possible that some of the other references might also be eliminated in this way.

The second and greater difficulty is availability of machines. The computation of a preliminary orbit and search ephemeris for a comet is usually done by an observer within a day or two of obtaining the third observation and indeed, if it is not done quickly, there is little point in doing it at all, and few observers are likely to have the necessary quick access to a large computer. The further stages of the problem, however, can be done much later since their object is to provide data for the detection of the body on its return. The application of high-speed computation to the calculation of preliminary orbits and search ephemerides is probably of only academic interest but, given a suitable organization, the problems arising in the calculation of general orbits and perturbations appear well suited to this type of treatment.

A most interesting calculation which shows the potentialities of digital computers in astronomical calculations has recently been completed in America on the I.B.M. machine. It was done by Dr. W. J. Eckert for the United States Nautical Almanac Office.

The positions of the five outer planets were calculated and tabulated from 1653 to 2060 at intervals of 40 days. Calculations were based on 25,000 observations made between 1780 and 1940 and on a few observations of Jupiter's satellites which go back as far as 1653. All the mutual interactions of the five planets and the sun were included in the calculations and the resultant set of 30 simultaneous non-linear differential equations was solved numerically to fourteen decimal places. This direct numerical approach to the problem had long been known, but it had been impracticable in the past because of the magnitude of the numerical work involved. The volume of published results contains one and a half million figures but these are less than 1 per cent of those which were produced and stored in the machine during the calculation. An elaborate checking procedure was required to verify the five million multiplications and seven million additions and subtractions which were done during the calculations, as a single error would have invalidated the whole of the work.

Chapter 25

DIGITAL COMPUTERS APPLIED TO GAMES

Chess problems are the hymn tunes of mathematics—G. H. Hardy

Machines which will play games have a long and interesting history. Among the first and most famous was the chess-playing automaton constructed in 1769 by the Baron Kempelen; M. Maelzel took it on tour all over the world, deceiving thousands of people into thinking that it played the game automatically. This machine was described in detail by Edgar Allan Poe; it is said to have defeated Napoleon himself—and he was accounted quite a good player, but it was finally shown up when somebody shouted "Fire" during a game, and caused the machine to go into a paroxysm owing to the efforts of the little man inside to escape.

In about 1890 Signor Torres Quevedo made a simple machine—a real machine this time—which with a rook and king can checkmate an opponent with a single king. This machine avoids stalemate very cleverly and always wins its games. It allows an opponent to make two mistakes before it refuses to play further with him, so it is always possible to cheat by moving one's own king the length of the board. The mechanism of the machine is such that it cannot move its rook back past its king and one can then force a draw! This machine, like Babbage's "noughts and crosses" machine is relatively simple, the rules to be obeyed are quite straightforward, and the machines couldn't lose. Babbage thought that his analytical engine ought to be able to play a real game of chess, which is a much more difficult thing to do.

In this chapter we discuss how a digital computer can be made to play chess—it does so rather badly, and how it plays draughts—it does so quite well. We shall also describe a special simple machine which was built to entertain the public during the Festival of Britain. It was called Nimrod because it played nim, a game which is like noughts and crosses, in that the tricks which are needed to win can be expressed in mathematical terms. This machine was on show in South Kensington for six months and took on all comers.

During the Festival the Society for Psychical Research came and fitted up a room nearby in order to see if the operations of the machine could be influenced by concentrated thought on the part of the research workers, most of whom were elderly ladies. When this experiment had failed they tried to discover whether they in turn could be affected by vibrations from the machine, and could tell from another room how the game was progressing. Unfortunately this experiment, like the first, was a complete failure, the only conclusion being that machines are much less co-operative than human beings in telepathic experiments.

At the end of the Festival of Britain Nimrod was flown to Berlin and shown at the Trade Fair. The Germans had never seen anything like it, and came to see it in their thousands, so much so in fact that on the first day of the show they entirely ignored a bar at the far end of the room where free drinks were available, and it was necessary to call out special police to control the crowds. The machine became even more popular after it had defeated the Economics Minister, Dr. Erhardt, in three straight games. After this it was taken to Canada and demonstrated to the Society of Engineers in Toronto. It is still somewhere on the North American continent, though it may have been dismantled by now, and it would be amusing to match it against some of the other nim-playing machines which have been built in the last year or two.

The reader might well ask why we bother to use these complicated and expensive machines in so trivial a pursuit as playing games. It would be disingenuous of us to disguise the fact that the principal motive which prompted the work was the sheer fun of the thing, but nevertheless if ever we had to justify the time and effort (and we feel strongly that no excuses are either necessary or called for) we could quite easily make a pretence at doing so. We have already explained how hard all programming is to do, and how much difficulty is due to the incompetence of the machine at taking an overall view of the problem which it is analysing. This particular point is brought out more clearly in playing games than in anything else. The machine cannot look at the whole of a chess board at once; it has to peer short-sightedly at every square in turn, in much the same way as it has to look at a commercial document. Research into the techniques of programming a machine to tackle complicated problems of this type may in fact lead to quite important advances, and help in serious work in business and economics—perhaps, regrettably, even in the theory of war. We hope that mathematicians will continue to play draughts and chess, and to enjoy themselves as long as they can.

We have often been asked why we don't use the machine to work out the football pools, or even to do something to remove the present uncertainty about the results of tomorrow's horse races. Perhaps one day we shall persuade our mathematicians to apply themselves to this problem too. It would first be necessary to establish a series of numerical criteria from which the machine could predict the results with greater certainty than the ordinary citizen can achieve with his pin; the presumption underlying the whole idea is that such criteria do in fact exist, but that they are too complicated for a man to apply in time, whereas a machine could do the necessary computations for him. It is most unlikely that a machine could ever hope to predict (for example) the results of a single football match, but it is at least possible that a detailed analysis might significantly improve the odds in favour of the gambler, so that if he invested on a large enough scale he could make a profit. It is notoriously true that mathematics, and particularly the theory of probability, owes more to gambling than gambling owes to mathematics; perhaps a machine might do something to restore the balance. Lady Lovelace lost a fortune by trying to back horses scientifically, and many others have done the same; all one could hope for is a slight improvement in the odds. We might make it pay but we doubt it; as an academic exercise it would be amusing, but we shall give the project a low priority.

CHESS

When one is asked, "Could one make a machine to play chess?" there are several possible meanings which might be given to the words. Here are a few—

(*a*) Could one make a machine which would obey the rules of chess, i.e. one which would play random legal moves, or which could tell one whether a given move is a legal one?

(*b*) Could one make a machine which would solve chess problems, e.g. tell one whether, in a given position, white has a forced mate in three?

(*c*) Could one make a machine which would play a reasonably good game of chess, i.e. which, confronted with an ordinary (that is, not particularly unusual) chess position, would after two or three minutes of calculation, indicate a passably good legal move?

(*d*) Could one make a machine to play chess, and to improve its play, game by game, profiting from its experience?

To these we may add two further questions, unconnected with chess, which are likely to be on the tip of the reader's tongue.

(*e*) Could one make a machine which would answer questions put to it, in such a way that it would not be possible to distinguish its answers from those of a man?

(*f*) Could one make a machine which would have feelings as you and I have?

The problem to be considered here is (*c*), but to put this problem into perspective with the others I shall give the very briefest of answers to each of them.

To (*a*) and (*b*) I should say, "This certainly can be done. If it has not been done already it is merely because there is something better to do."

Question (*c*) we are to consider in greater detail, but the short answer is, "Yes, but the better the standard of play required, the more complex the machine must be, and the more ingenious perhaps the designer."

To (*d*) and (*e*) I should answer, "I believe so. I know of no really convincing argument to support this belief, and certainly of none to disprove it."

To (*f*) I should say, "I shall never know, any more than I shall ever be quite certain that *you* feel as I do."

In each of these problems except possibly the last, the phrase, "Could one make a machine to . . ." might equally well be replaced by, "Could one programme an electronic computer to . . ." Clearly the electronic computer so programmed would itself constitute a machine. And on the other hand if some other machine had been constructed to do the job we could use an electronic computer (of sufficient storage capacity), suitably programmed, to calculate what this machine would do, and in particular what answer it would give.

After these preliminaries let us give our minds to the problem of making a machine, or of programming a computer, to play a tolerable game of chess. In this short discussion it is of course out of the question to provide actual programmes, but this does not really matter on account of the following principle—

If one can explain quite unambiguously in English, with the aid of mathematical symbols if required, how a calculation is to be done, then it is always possible to programme any digital computer to do that calculation, provided the storage capacity is adequate.

This is not the sort of thing that admits of clear-cut proof, but amongst workers in the field it is regarded as being clear as day. Accepting this principle, our problem is reduced to explaining "unambiguously in English" the rules by which the machine is to choose its move in each position. For definiteness we will suppose the machine is playing white.

If the machine could calculate at an infinite speed, and also had unlimited storage capacity, a comparatively simple rule would suffice, and would give a result that in a sense could not be improved on. This rule could be stated:

"Consider every possible continuation of the game from the given position. There is only a finite number of them (at any rate if the fifty-move rule makes a draw obligatory, not merely permissive). Work back from the end of these continuations, marking a position with white to play as 'win' if there is a move which turns it into a position previously marked as 'win.' If this does not occur, but there is a move which leads to a position marked 'draw,' then mark the position 'draw.' Failing this, mark it 'lose.' Mark a position with black to play by a similar rule with 'win' and 'lose' interchanged. If after this process has been completed it is found that there are moves which lead to a position marked 'win,' one of these should be chosen. If there is none marked 'win' choose one marked 'draw' if such exists. If all moves lead to a position marked 'lose,' any move may be chosen."

Such a rule is practically applicable in the game of noughts and crosses, but in chess is of merely academic interest. Even when the rule can be applied it is not very appropriate for use against a weak opponent, who may make mistakes which ought to be exploited.

In spite of the impracticability of this rule it bears some resemblance to what one really does when playing chess. One does not follow all the continuations of play, but one follows some of them. One does not follow them until the end of the game, but one follows them a move or two, perhaps more. Eventually a position seems, rightly or wrongly, too bad to be worth further consideration, or (less frequently) too good to hesitate longer over. The further a position is from the one on the board the less likely it is to occur, and therefore the shorter is the time which can be assigned for its consideration. Following this idea we might have a rule something like this—

"Consider all continuations of the game consisting of a move by white, a reply by black, and another move and reply. The value of the position at the end of each of these sequences of moves is estimated according to some suitable rule. The values at earlier positions are then calculated by working backwards move by move as in the theoretical rule given before. The move to be chosen is that which leads to the position with the greatest value."

It is possible to arrange that no two positions have the same value. The rule is then unambiguous. A very simple form of values, but one not having this property, is an "evaluation of material," e.g. on the basis—

$$\begin{aligned} P &= 1 \\ Kt &= 3 \\ B &= 3\tfrac{1}{2} \\ R &= 5 \\ Q &= 10 \\ \text{Checkmate} &= 1000 \end{aligned}$$

If B is black's total and W is white's, then W/B is quite a good measure of value. This is better than $W - B$ as the latter does not encourage exchanges when one has the advantage. Some small extra arbitrary function of position may be added to ensure definiteness in the result.

The weakness of this rule is that it follows all combinations equally far. It would be much better if the more profitable moves were considered in greater detail than the less. It would also be desirable to take into account more than mere "value of material."

After this introduction I shall describe a particular set of rules, which could without difficulty be made into a machine programme. It is understood that the machine is white and that white is next to play. The current position is called the *position on the board*, and the positions arising from it by later moves *positions in the analysis*.

"Considerable" Moves

"Considerable" here is taken to mean moves which will be "considered" in the analysis by the machine.

Every possibility for white's next move and for black's reply is "considerable." If a capture is considerable then any recapture is considerable. The capture of an undefended piece or the capture of a piece of higher value by one of lower value is always considerable. A move giving checkmate is considerable.

Dead Position

A position in the analysis is dead if there are no considerable moves in that position, i.e. if it is more than two moves ahead of the present position, and no capture or recapture or mate can be made in the next move.

Value of Position

The value of a dead position is obtained by adding up the piece values as above, and forming the ratio W/B of white's total to black's. In other positions with white to play the value is the greatest value of (*a*) the positions obtained by considerable moves, or (*b*) the position itself evaluated as if a dead position. The latter alternative is to be omitted if all moves are considerable. The same process is to be undertaken for one of black's moves, but the machine will then choose the *least* value.

Position-play Value

Each white piece has a certain position-play contribution and so has the black king. These must all be added up to give the position-play value.

For a Q, R, B, or Kt, count—

(*a*) The square root of the number of moves the piece can make from the position, counting a capture as two moves, and not forgetting that the king must not be left in check.

(*b*) (If not a Q) 1·0 if it is defended, and an additional 0·5 if twice defended.

For a K, count—

(*c*) For moves other than castling as (*a*) above.

(*d*) It is then necessary to make some allowance for the vulnerability of the K. This can be done by assuming it to be replaced by a friendly Q on the same square, estimating as in (*a*), but subtracting instead of adding.

(*e*) Count 1·0 for the possibility of castling later not being lost by moves of K or rooks, a further 1·0 if castling could take place on the next move, and yet another 1·0 for the actual performance of castling.

For a P, count—

(*f*) 0·2 for each rank advanced.

(*g*) 0·3 for being defended by at least one piece (not P).

For the black K, count—

(*h*) 1·0 for the threat of checkmate.

(*i*) 0·5 for check.

We can now state the rule for play as follows. The move chosen must have the greatest possible value, and, consistent with this, the greatest possible position-play value. If this condition admits of several solutions a choice may be made at random, or according to an arbitrary additional condition.

Note that no "analysis" is involved in position-play evaluation. This is to reduce the amount of work done on deciding the move.

The game below was played between this machine and a weak player who did not know the system. To simplify the calculations the square roots were rounded off to one decimal place, i.e. this table was used—

Number . . .	0	1	2	3	4	5	6	7	8	9	10	11	12	13
Square Root . .	0	1	1·4	1·7	2·0	2·2	2·4	2·6	2·8	3·0	3·2	3·3	3·5	3·6

No random choices actually arose in this game. The increase of position-play value is given after white's move if relevant. An asterisk indicates that every other move had a lower position-play value.

	White (*Machine*)		*Black*
1.	P—K 4	4·2*	P—K 4
2.	Kt—Q B 3	3·1*	Kt—K B 3
3.	P—Q 4	2·6*	B—Q Kt 5
4.	Kt—K B 3 (1)	2·0	P—Q 3
5.	B—Q 2	3·5*	Kt—Q B 3
6.	P—Q 5	0·2	Kt—Q 5
7.	P—K R 4 (2)	1·1*	B—Kt 5
8.	P—Q R 4 (2)	1·0*	Kt×Kt ch.
9.	P×Kt		B—K R 4
10.	B—Kt 5 ch.	2·4*	P—Q B 3
11.	P×P		O—O
12.	P×P		R—Kt 1
13.	B—R 6	−1·5	Q—R 4
14.	Q—K 2	0·6	Kt—Q 2
15.	K R—Kt 1 (3)	1·2*	Kt—B 4 (4)
16.	R—Kt 5 (5)		B—Kt 3
17.	B—Kt 5	0·4	Kt×Kt P
18.	O—O—O	3·2*	Kt—B 4
19.	B—B 6		K R—Q B 1
20.	B—Q 5		B×Kt
21.	B×B	0·7	Q×P
22.	K—Q 2		Kt—K 3
23.	R—Kt 4	−0·3	Kt—Q 5
24.	Q—Q 3		Kt—Kt 4
25.	B—Kt 3		Q—R 3
26.	B—B 4		B—R 4
27.	R—Kt 3		Q—R 5
28.	B×Kt		Q×B
29.	Q×P (6)		R—Q 1 (4)
30.	Resigns (7)		

Notes—

1. If B—Q 2 3·7* then P×P is foreseen.
2. Most inappropriate moves.
3. If white castles then B×Kt, B×B, Q×P.
4. The fork is unforeseen at white's last move.
5. Heads in the sand!
6. Fiddling while Rome burns!
7. On the advice of his trainer.

Numerous criticisms of the machine's play may be made. It is quite defenceless against forks, although it may be able to see certain other kinds of combination. It is of course not difficult to devise improvements of the programme so that these simple forks are foreseen. The reader may be able to think of some such improvements for himself. Since no claim is made that the above rule is particularly good, I have been content to leave this flaw without remedy; clearly a line has to be drawn between the flaws which one will attempt to eliminate and those which must be accepted as a risk. Another criticism is that the scheme proposed, although reasonable in the middle game, is futile in the end game. The change-over from the middle game to the end-game is usually sufficiently clear-cut for it to be possible to have an entirely different system for the end-game. This should of course include quite definite programmes for the standard situations, such as mate with rook and king, or king and pawn against king. There is no intention to discuss the end-game further here.

If I were to sum up the weakness of the above system in a few words I would describe it as a caricature of my own play. It was in fact based on an introspective analysis of my thought processes when playing, with considerable simplifications. It makes oversights which are very similar to those which I make myself, and which may in both cases be ascribed to the considerable moves being inappropriately chosen. This fact might be regarded as supporting the glib view which is often expressed, to the effect that "one cannot programme a machine to play a better game than one plays oneself." This statement should I think be compared with another of rather similar form. "No animal can swallow an animal heavier than himself." Both statements are, as far as I know, untrue. They are also both of a kind that one is easily bluffed into accepting, partly because one thinks that there ought to be some slick way of demonstrating them, and one does not like to admit that one does not see what this argument is. They are also both supported by normal experience, and need exceptional cases to falsify them. The statement about chess programming may be falsified quite simply by the speed of the machine, which might make it feasible to carry the analysis a move farther than a man could do in the same time. This effect is less than might be supposed. Although electronic computers are very fast where conventional computing is concerned, their advantage is much reduced where enumeration of cases, etc., is involved on a large scale. Take for instance the problem of counting the possible moves from a given position in chess. If the number is 30 a man might do it in 45 seconds and the machine in 1 second. The machine has still an advantage, but it is much less overwhelming than it would be for instance when calculating cosines.

In connexion with the question of the ability of a chess-machine to profit from experience, one can see that it would be quite possible to programme the machine to try out variations in its method of play (e.g. variations in piece value) and adopt the one giving the most satisfactory results. This could certainly be described as "learning," though it is not quite representative of learning as we know it. It might also be possible to programme the machine to search for new types of combination in chess. If this project produced results which were quite new, and also interesting to the programmer, who should have the credit? Compare this with the situation where a Defence Minister gives orders for research to be done to find a counter to the bow and arrow. Should the inventor of the shield have the credit, or should the Defence Minister?

The Manchester University Machine

In November, 1951, some months after this article was written (by Dr. Turing) Dr. Prinz was able to make the Manchester University machine solve a few straightforward chess problems of the "Mate-in-Two" type (see *Research*, Vol. 6 (1952), p. 261).

It is usually true to say that the best and often the only way to see how well the machine can tackle a particular type of problem is to produce a definite programme for the machine, and, in this case, in order to have something working in the shortest possible time, a few restrictions were imposed on the rules of chess as they were "explained" to the machine. For example castling was not permitted, nor were double moves by pawns, nor taking *en passant* nor the promotion of a pawn into a piece when it reached the last row; further, no distinction was made between mate and stalemate.

The programme contained a routine for the construction of the next possible move, a routine to check this move for legality, and various sequences for recording the moves and the positions obtained. All these separate subroutines were linked together by a master routine which reflected the structure of the problem as a whole and ensured that the subroutines were entered in the proper sequence.

The technique of programming was perhaps rather crude, and many refinements, increasing the speed of operation, are doubtless possible. For this reason, the results reported here can only serve as a very rough guide to the speed attainable; but they do show the need for considerable improvement in programming technique and machine performance before a successful game by a machine against a human chess player becomes a practical possibility.

The programme, as well as the initial position on the chess board, was supplied to the machine on punched tape and then transferred to the magnetic store of the machine.

An initial routine (sub-programme) was transferred to the electronic store, and the machine started its computation. The programme was so organized that every first move by white was printed out; after the key move had been reached the machine printed: "MATE."

The main result of the experiment was that the machine is disappointingly slow when playing chess—in contrast to the extreme superiority over human computers where purely mathematical problems are concerned. For the simple example given in the position reproduced here, 15 minutes were needed to print the solution. A detailed analysis shows that the machine tried about 450 possible moves (of which about 100 were illegal) in the course of the game; this means about two seconds per move on the average.

A considerable portion of this time had to be used for a test for self-check (i.e. after a player had made a move, to find out whether his own King was left in check). This was done by first examining all squares connected to the King's square by a Knight's move, to see (*a*) whether they were on the board at all, (*b*) whether they were empty or occupied, (*c*) if occupied, by a piece of which colour and (*d*) if occupied by a piece of opposite colour, whether or not this piece was a Knight. A similar test had to be carried out for any other piece that might have put the King in check. This test involves several hundreds of operations and, at a machine speed of 1 msec per operation, might take an appreciable fraction of a second.

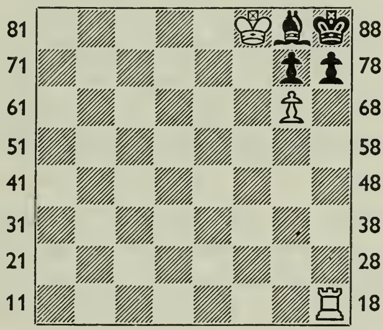
The next important time-consuming factor was the magnetic transfers, i.e. the transfers of sub-programmes and data (relating to positions and moves) between the magnetic and the electronic store. It is here that improved programming technique may save time by better utilization of the electronic store, thus reducing the number of transfers (nine for every legal move in the present programme).

Compared with these two items, the time spent in computing the moves appeared to be of minor importance although the machine not only computed the possible moves but also the impossible, but "thinkable" moves—meaning those which either carry the piece off the board, or lead to a collision with a piece of the same colour already on the square. These moves, however, were quickly rejected by the machine and did not contribute greatly to the total computation time.

It appears that if this crude method of programming were the only one available it would be quite impractical for any machine to compete on reasonable terms with a competent human being.

Before we conclude too easily that no computer will ever compete in a Masters' Tournament let us remind ourselves that the Manchester machine solved a problem after a few weeks tuition, which represents quite reasonable progress for a beginner.

The First Chess Problem Solved by a Computing Machine. The task set the Manchester machine was to find a move by white that would lead to a mate in the next move, whatever black might answer. The move is R—R6.



For solution of the problem by the machine the squares of the board were numbered in rather unusual fashion. The bottom row was numbered 11 to 18 (from left to right), the next 21 to 28, and so on to the top row, which was 81–88. Square 68 was thus the square in row 6, column 8. The machine has printed out all the moves which white tried out to find a solution, and has printed "MATE" after finding and recording the key move, which appears in the form "Rook to 68."

The list of moves is—

Pawn to 78.
Rook to 17.
Rook to 16.
Rook to 15.
Rook to 14.
Rook to 13.
Rook to 12.
Rook to 11.
Rook to 28.
Rook to 38.
Rook to 48.
Rook to 58.
Rook to 68.
MATE.

DRAUGHTS

The game of draughts occupies an intermediate position between the extremely complex games such as chess, and the relatively simple games such as nim or noughts-and-crosses for which a complete mathematical theory exists. This fact makes it a rather suitable subject for experiments in mechanical game playing, for although there is no complete theory of the game available, so that the machine has to look ahead to find the moves, the moves themselves are rather simple and relatively few in number.

Various forms of strategy have been suggested for constructing an automatic chess player; the purpose of such plans is to reduce the time taken by the machine to choose its move. As Prinz has shown, the time taken by any machine which considers all the possible moves for four or five steps ahead would be quite prohibitive, and the principal aim of the strategy is to reduce this number very considerably, while at the same time introducing a scheme of valuing the positions which will allow it to choose a reasonably good move. The chief interest in games-playing machines lies in the development of a suitable strategy.

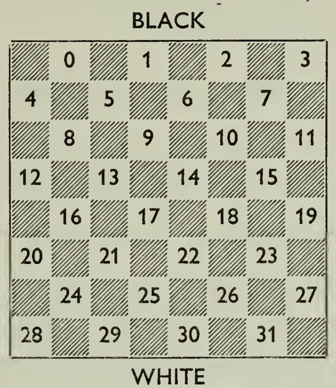
Before any strategy can be realized in practice, however, the basic programme necessary to find the possible moves and to make them must be constructed. When this has been done the strategy, which consists principally of the methods by which positions can be valued, can be added to make the complete game player. It is obviously possible to make experiments with different strategies using the same basic move-finding-and-making routine.

The basic programme for draughts, which is described in outline in the following paragraphs, is very much simpler than the corresponding one for chess. It has in fact proved possible to put both it and the necessary position storage in the electronic store of the Manchester machine at the same time. This removes the need for magnetic transfers during the operation of the programme, and this fact, together with the simplicity of the moves, has reduced the time taken to consider a single move to about one tenth of a second.

Basic Programme for Draughts

We must first consider the representation of a position in the machine. The 32 squares used in a draughts board are numbered as shown in the diagram.

A position is represented by 3 thirty-two-digit binary numbers (or words) B, W and K which give the positions of the black men (and kings), the white men (and kings) and the kings (of either colour) respectively. The digits of these words each represent a square on the board; the square n being represented by the digit 2^n.



Thus the least significant digit represents square 0 and the most significant digit represents square 31. (In the Manchester machine, where the word length is 40 digits, the last 8 digits are irrelevant.) A unit in the word indicates the presence, and a zero indicates the absence of the appropriate type of man in the corresponding square. Thus the opening position of the game would be represented by*—

$B =$ 1111, 1111, 1111, 0000, 0000, 0000, 0000, 0000

$W =$ 0000, 0000, 0000, 0000, 0000, 1111, 1111, 1111

$K =$ 0000, 0000, 0000, 0000, 0000, 0000, 0000, 0000

The positions of the white kings are indicated by the word $W\&K$, while the empty squares are indicated by the word $\sim W \& \sim B$.†

It will be seen that there are at most four possible types of non-capture moves from any square on the board. For example, from square 14 the possible moves are to squares 9, 10, 17 or 18. The machine considers all these moves in turn, but it will be sufficient to indicate here the way in which it deals with one of them—say the move 14–18.

* All binary numbers are written in the convention used for the Manchester machine, i.e. with their *least* significant digit on the left.

† $W\&K$ stands for the logical product of W and K (sometimes also known as the result of collating W and K). $\sim W$ stands for the negation of W, i.e. the word obtained by writing 1's for 0's in W, and vice versa (see Chapter 15).

This type of move, which consists of adding 4 to the number of the square, corresponds to multiplying the appropriate digit in the position word by 2^4. A move of this type can be made by any black man, but only by a white king; it cannot be made from squares 28, 29, 30 or 31 nor can it be made unless the square to which the man is to be moved is empty. For a black move, the machine therefore forms the following quantity—

$$Y = \{(B\&M) \times 2^4\} \&{\sim}W\&{\sim}B$$

where $M =$ 1111, 1111, 1111, 1111, 1111, 1111, 1111, 0000
For a white move, the corresponding quantity is—

$$\{(W\&K\&M) \times 2^4\} \&{\sim}W\&{\sim}B$$

In these expressions $(B\&M)$ or $(W\&K\&M)$ give all the men on the board who could make the move; multiplying this by 2^4 give the squares to which they would move. If these squares are empty (collate with ${\sim}W\&{\sim}B$) the move is possible.

The quantity Y thus represents all the possible moves of this type. To consider a single one of these, the largest non-zero digit of Y is taken and removed from Y. The word consisting of this single digit known as θ, gives the square to which the man is moved. The quantity $\phi = \theta \times 2^{-4}$ is then formed and gives the square from which the man was moved. For a black move, the quantity—

$$B' = B \not\equiv \theta \not\equiv \phi$$

will then give the new position of the black men. If $K\&\phi$ is not zero, the man moved was a king so that $K' = K \not\equiv \theta \not\equiv \phi$ gives the new position of the kings. If $K\&\phi$ is zero, the man moved was not a king. The new position of the kings will therefore be unaltered unless the man has kinged during this move—in other words unless $\theta \geqslant 2^{28}$ in which case $K' = K \not\equiv \theta$.

Relatively simple modifications of this scheme are needed to deal with white moves and non-capture moves of other types. Capture moves are somewhat more complicated as multiple captures must be allowed for. Furthermore, all the possible captures must be made or the machine will render itself liable to be huffed. This leads to a considerable complication which it is not possible to describe fully here, but the basic scheme is not altered.

The machine considers all the possible moves of one type before starting the next, so that in order to describe a position fully, it is necessary to store the word *Y*, which indicates the moves still to be considered, as well as the position words *B*, *W* and *K*. It is also necessary to keep a record of the type of move being considered. This is done with the aid of a further parameter word *P* which also contains the value associated with the position. The whole storage required for a position is thus reduced to the 5 thirty-two-digit words *B*, *W*, *K*, *Y*, and *P*.

Valuation of Positions and Strategy

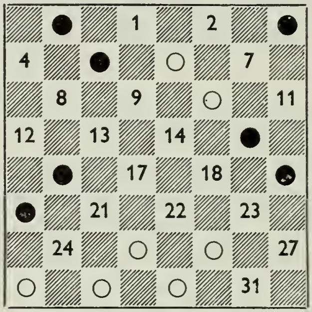
It should be possible to graft almost any type of strategy on to the move-finding scheme outlined above to produce a complete draughts-playing routine and then to evaluate the effectiveness of the strategy by direct experiment. I have done this with two rather simple types of strategy so far, and I hope to be able to try some rather more refined strategies in the future.

For demonstration purposes, and also to ensure that a record of the game is kept, and to take certain precautions against machine error, the move-finding sequence and the associated strategy have been combined with a general game-playing routine which accepts the opponent's moves, displays the positions, prints the move, and generally organizes the sequence of operations in the game. It is rather typical of logical programmes that this organizing routine is in fact longer than the game-playing routine proper. As its operations, though rather spectacular, are of only trivial theoretical interest, I shall not describe them here.

The first, and simplest, strategy to try is the direct one of allowing the machine to consider all the possible moves ahead on both sides for a specified number of stages. It then makes its choice, valuing the final resulting positions only in terms of the material left on the board and ignoring any positional advantage. There is an upper limit to the number of stages ahead that can be considered owing to limitations of storage space—actually six moves, three on each side, are all that can be allowed. In practice, however, time considerations provide a more severe limitation. There are on an average about ten possible legal moves at each stage of the game, so that consideration of one further stage multiplies the time for making the move by a factor of about ten. The machine considers moves at the rate of about ten a second, so that looking three moves ahead (two of its own and one of its opponents), which takes between one and two minutes, represents about the limit which can be allowed from the point of view of time.

This is not sufficient to allow the machine to play well, though it can play fairly sensibly for most of the game. One wholly unexpected difficulty appears. Consider the position on the following board.

BLACK



WHITE

In this position, the machine (black) is aware that its opponent is going to king next move. Now a king is more valuable than a man—the actual values used were three for a king and one for a man—so that if the opponent kings the machine effectively loses two points. The only way it can stop this is by offering a man for sacrifice, because then, by the rules of the game, the sacrifice must be taken at once. If the machine does this, it will lose only one point, and as it is not looking far enough ahead, it cannot see that it has not prevented its opponent from kinging but only postponed the evil day. At its next move it is still faced with the same difficulty, which it tries to solve in the same way, so that it will make every possible sacrifice of a single man before it accepts as inevitable the creation of an opponent's king. In fact, when faced with this position, the machine played 19—23, followed by 16—21 and 20—24.

This, of course, is a fatal flaw in the strategy—and not one it would have been easy to discover without actually trying it out. An opponent who detected this behaviour—and it is extremely conspicuous in play—would only have to leave his man on the point of kinging indefinitely. The machine would then sacrifice all its remaining men as soon as the opportunity offered.

In order to avoid this difficulty, the second strategy was devised. In this the machine continues to investigate the moves ahead until it has found two consecutive moves without captures. This means that it will be able to recognize the futility of its sacrifice to prevent kinging. It is still necessary to impose an over-riding limit on the number of stages it can consider, and once more, considerations of time limit this. However, as no move is continued for more than two stages unless it leads to a capture, it is possible to allow the machine to consider up to four stages ahead without it becoming intolerably slow. This would mean that it would consider the sacrifice of two men to be of equal value to the creation of an

	Machine	*Strachey*
1.	11—15	23—18
2.	7—11	21—17
3.	8—12	20—16(1)
4.	12—21(16)	25—16(21)
5.	9—14!(2)	18—9(14)
6.	6—20(16,9)(3)	27—23
7.	2—7(4)	23—18
8.	5—8	18—14
9.	8—13(5)	17—8(13)
10.	4—13(8)	14—9
11.	1—5(6)	9—6
12.	15—19	6—1 K
13.	5—9	1—6?(7)
14.	0—5!(8)	6—15(10)
15.	11—25(22,15)	30—21(25)
16.	13—17	21—14(17)
17.	9—18(14)	24—21
18.	18—23	26—22
19.	23—27	22—17
20.	5—8(9)	17—14
21.	8—13	14—9
22.	19—23	9—6
23.	23—26(10)	31—22(26)
24.	27—31 K	6—2 K
25.	7—10	2—7
26.	10—15	21—16?(11)
27.	3—10(7)	16—9(13)
28.	10—14	9—6
29.	15—19	6—2 K
30.	31—27(12)	2—6
31.	27—31(12)	6—10
32.	31—26(13)	10—17(14)
33.	19—23	29—25
34.	26—31(14)	

Notes—

1. An experiment on my part—the only deliberate offer I made. I thought, wrongly, that it was quite safe.
2. Not foreseen by me.
3. Better than 5—21(9,17).
4. A random move (zero value). Shows the lack of a constructive plan.
5. Another random move of zero value, actually rather good.
6. Bad. Ultimately allows me to make a King. 10—14 would have been better.
7. A bad slip on my part.
8. Taking full advantage of my slip.
9. Bad. Unblocks the way to a King.
10. Sacrifice in order to get a King (not to stop me kinging). A good move, but not possible before 19—23 had been made by chance.
11. Another bad slip on my part.
12. Purposeless. The strategy is failing badly in the end game.
13. Too late.
14. Futile. The game was stopped at this point as the outcome was obvious.

opponent's king, and as there is a random choice between moves of equal value, it might still make this useless sacrifice. This has been prevented by reducing the value of a king from 3 to $2\frac{7}{8}$.

With this modified strategy, the machine can play quite a tolerable game until it reaches the end game. It has always seemed probable that a wholly different strategy will be necessary for end games. The game given on page 303, which is the first ever played using the strategy, brings this point out very clearly.

NIM

A considerably easier game which the machine can be programmed to play is the one known as nim. Probably a variation of this was known to the Chinese—certainly in its present form many people have met it. We have chosen to deal with this comparatively trivial game in detail because of its topical interest. Thousands of people will have seen *Nimrod*, the computer built by Ferranti Ltd. for the Science Exhibition of the Festival of Britain. This special-purpose machine was designed to show the main features of large electronic digital computers, and the game of nim was chosen as an interesting but simple demonstration problem. The game itself is as follows—

Initially we have any number of heaps, each containing any number of tokens (usually matches). In the simplest form, two contestants play alternately, and may pick up as many matches as they wish at one time from *one* pile, but they must take at least one match. The aim is to avoid taking the last match of all—or there is another variation where the aim is to take the last match or group of matches.

The so-called *multiple game* differs from this only in that the number of heaps altered in any move may take any value from one up to a pre-assigned maximum k. Of course, to prevent complete triviality, k must be less than N, the total number of heaps.

The detailed theory of nim was worked out long ago and, apart from the initial distribution of the matches, no element of chance need enter into the game. This theory is very simple, but it becomes clearer for the non-mathematician if we use the concept of a binary number, introduced elsewhere (see page 33).

We can now proceed to give a working rule for the game of nim. We would like to find a *winning position* having the following characteristics—

(*a*) It is impossible, when faced by a winning position, to make a move which will leave a winning position.

(*b*) Faced with any other than a winning position, it is possible to make a move resulting in a winning position.

(*c*) If at any stage of the game a player *A* can convert a position into a winning position, it is possible for *A* to win, and impossible for his opponent *B* to do so unless *A* makes a mistake. *A* wins by leaving a winning position at every succeeding move on his part.

Such winning positions can be achieved and are recognized as follows: For any given configuration, express the number of matches in each heap as a binary number. Suppose, for example, that we have four heaps, *A*, *B*, *C* and *D*, containing respectively 7, 4, 3 and 2 matches. These are represented—

	4	2	1	
A	1	1	1	(7)
B	1	0	0	(4)
C	0		1	(3)
D	0		0	(2)

We write these down as above, one under the other, and add up each column, e.g., in the above example, we get

	4	2	1
Sum:	2	3	2

Now the "secret" of a winning position is that every column should be divisible by $k + 1$; k being the maximum number of heaps which can be altered in any one move. Thus the example quoted above cannot represent a winning position whatever our initial choice of k. However, suppose we have $k = 1$; then consider the position—

	4	2	1	
A	1	0	1	(5)
B	1	1	1	(7)
C	0	1	1	(3)
D	0	0	1	(1)
Sum:	2	2	4	

This is a winning position, but would not be so if we had previously fixed $k = 3$, for example.

To convert an "unsafe" into a winning position, we deal with a column at a time. Consider our previous example with $k = 1$.

	4	2	1	
A	1	1	1	(7)
B	1	0	0	(4)
C	0	1	1	(3)
D	0	1	0	(2)
Sum:	2	3	2	

We start with the "most-significant," or left-hand column. This sum is divisible by $k + 1$, so we proceed to consideration of the next column. The sum here is 3, which is not divisible by $k + 1$, so we choose any heap, say *D*, having a one in this column. We remove this 1 (which is equivalent to subtracting 2 from *D*), and put 1 in every less-significant (or right-hand) column of this heap (which in this case is equivalent to adding 1, though if we had chosen to modify *A* instead, it would have meant no change in the last column). That is, we make the minimum move which removes the 1 in the "unsafe" column. Thus we remove 1 from *D*, and so alter its binary representation to 001.

Now our representation is—

	4	2	1	
A	1	1	1	(7)
B	1	0	0	(4)
C	0	1	1	(3)
D	0	0	1	(1)
Sum:	2	2	3	

and we see that we have made the sum of column 2 divisible by $k + 1$ at the expense of column 1. However, we shall now proceed to adjust column 1. To avoid altering more than k heaps in one move, we must alter one or more of the heaps already affected if, by so doing, we can achieve the desired result, rather than select a fresh heap.

Now, in this case, we wish to remove 1 from column 1 of some heap. Since heap *D* has already been altered, we choose this—it has a 1 in this column.

So, at the end of our move, we have removed two matches from heap *D*, and leave the winning position—

	4	2	1	
A	1	1	1	(7)
B	1	0	0	(4)
C	0	1	1	(3)
D	0	0	0	(0)
Sum:	2	2	2	

In adapting this game for the universal computer, we allow a maximum of eight heaps, with not more than thirty-one matches in a heap. In Nimrod the more stringent restrictions to four heaps, each with a maximum content of seven matches, were applied to simplify the problems of demonstration.

Possible positions with which the machine may be faced are as follows—

(*a*) At least $k + 1$ heaps contain more than one match.

(*b*) The number of heaps containing more than one match lies between 1 and k (inclusive).

(*c*) No heap contains more than one match. Not all heaps are empty.

(*d*) All heaps are empty.

In case (*a*), we follow the so-called *normal routine*, which aims at leaving column sums all divisible by $(k + 1)$.

In case (*b*), we want to leave $r\ (k + 1) + 1$ heaps containing one match, and no heaps with more than one, where r may have any non-negative integral value (i.e. $r = 0, 1, 2, \ldots$).

In case (*c*) the same applies. If only one heap is left, containing one match, we have no choice of move, but this need not be treated separately.

In case (*d*), the game is over. Special investigation has to be used to detect this case. In all other cases, if the normal routine cannot succeed in its purpose, i.e. if the machine is faced with a winning position—a random move can, and must, be made. But, in this situation, this obviously cannot be done.

Thus the routine breaks up naturally into the following parts—

(i) Entry
(ii) Determination of case
(iii) Normal Routine
(iv) Cases (*b*) and (*c*)
(v) Treatment of zero case (*d*)
(vi) Random move
(vii) Emergence.

There is no need to give further details of the programme, but an example is given of how the machine would tackle a specific game.

Suppose initially that we have four heaps, containing respectively 7, 4, 5 and 2 matches; that $k = 2$; and that the machine moves first.

(i) Entry—

	4	2	1	
A	1	1	1	(7)
B	1	0	0	(4)
C	1	0	1	(5)
D	0	1	0	(2)

(ii) Determination of case—

There are 4 non-zero, non-unit heaps, so we are dealing with case (*a*).

(iii) Normal routine—

	4	2	1	
A	1	1	1	(7)
B	1	0	0	(4)
C	1	0	1	(5)
D	0	1	0	(2)
Sum:	3	2	2	

The sum of column 4 is divisible by $k + 1$ so we need not modify it.

The sum of column 2 is 2, and is not divisible by $k + 1$, so we need to modify any heap having a 1 in this column—say heap A.

According to the rules, we then get—

	4	2	1	
A	1	0	1	(5)
B	1	0	0	(4)
C	1	0	1	(5)
D	0	1	0	(2)
Sum:	3	1	2	

And we note that heap A has been modified, and should be again modified whenever possible. Sum of column 2 is still not divisible by $k + 1$, so this time we modify heap D to obtain—

	4	2	1	
A	1	0	1	(5)
B	1	0	0	(4)
C	1	0	1	(5)
D	0	0	1	(1)
Sum:	3	0	3	

Column 2 is now divisible by $k + 1$ and, proceeding to the next column, we see this condition is also satisfied here, so the move has been completed and a winning-position left, the means to this end being the removal of two matches from A, and one from D, leaving 5, 4, 5 and 1. (If column 1 had needed adjustment, we should have had to modify one or both of heaps A and D, since these had already been affected.)

Suppose the opponent now makes a move leaving 0, 4, 2 and 1 as the contents of the respective heaps. It is now for the machine to move again.

(i) Entry—

	4	2	1	
A	0	0	1	(0)
B	1	0	0	(4)
C	0	1	0	(2)
D	0	0	1	(1)

(ii) Determination of case.

There are 3 non-zero, non-unit heaps, so we are dealing with case (*b*). Thus we want to leave 1, or 4, or 7 . . . unit-heaps. Clearly we can only leave 1 unit heap in this case.

(iv) Cases (*b*) and (*c*).

We remove all matches from heaps *B* and *D*, which affects only *k* heaps, and leaves just one unit heap as required.

The opponent is now forced to remove the last match, and the machine wins the game.

Chapter 26

THOUGHT AND MACHINE PROCESSES

Cogito, ergo sum—Descartes
I do not think, therefore I am not—Dr. Strabismus (whom God preserve) of Utrecht. President of the Anti-cartesian Society

So far we have described the construction of these new digital computers, and tried to show how useful they can become by doing routine computations. If we are to complete the story we must also try to assess the limitations of the machines which we can build today, and, if possible, to discuss any limits to the performance of machines which may be built in the future. We shall try to compare the processes which go on inside them with those which are responsible for the thoughts in our own minds. This subject is far too complicated to be dealt with in a single chapter, but we shall try to describe some of its more important aspects. We shall begin by giving an account of some of the astonishing feats of mental arithmetic which are demonstrated by those rare individuals who are known as "calculating prodigies." Many accounts of these men have appeared, of which one of the best known is to be found in W. W. R. Ball's book,* to which the reader is referred for a comprehensive historical account of the subject.

At rare intervals there have appeared men and boys who display extraordinary powers of mental arithmetic. In a few seconds they can give the answer to questions which an expert mathematician could obtain only in a much longer time with the aid of pencil and paper. Some of them have remained otherwise illiterate; others, such as Gauss, Ampère, and Bidder, have risen to positions of eminence as mathematicians, physicists, or engineers. Many of them seem to have taught themselves the rules of arithmetic in their childhood, and to have learnt the multiplication table by playing with pebbles. Few of these prodigies have been able to explain in detail how they achieve their apparently miraculous results, but two of the most remarkable of them have been kind enough to discuss their methods with us. We are therefore much indebted to Dr. A. C. Aitken, F.R.S., Professor of Mathematics in Edinburgh

* *Mathematical Recreations and Essays*. See also *Common Sense and Its Cultivation,* by Hanbury Hankin, and *Mental Prodigies,* by Fred Barlow.

University, to several of his old students, to Dr. Stokvis, and to Dr. van Wijngaarden and Mr. William Klein of the Mathematisch Centrum, Amsterdam, for the information which we have used in the following pages.

As far as we can judge, Professor Aitken and Mr. Klein use very similar methods in their mental computations; their speeds are quite comparable, and they are at least as fast as, and probably faster than, any of the prodigies whose performances have been described in the past.

Both men have most remarkable memories—they know by heart the multiplication table up to 100×100, all squares up to 1000×1000, and an enormous number of odd facts, such as $3937 \times 127 = 499999$, which are very useful to them, and seem to arise instantaneously in their minds when they are needed. In addition Mr. Klein knows by heart the logarithms of all numbers less than 100, and of all prime numbers less than 10,000 (to twenty decimal places) so that he can work out sums like compound interest by "looking up" the logs in his head, after factorizing the numbers he is using, if need be. He has also learnt enough about the calendar to be able to give the day of the week corresponding to any specified date in history; he learnt most of the Amsterdam telephone directory for fun. Professor Aitken has neglected logarithms in favour of mathematical formulae and the piano music and violin sonatas of Bach and Beethoven, but nevertheless he learnt 802 places of π by heart in about fifteen minutes, an operation which to him was comparable in difficulty to learning a Bach fugue.

If one realizes that in addition to their phenomenal memories both men possess an equally phenomenal ability at mental arithmetic, one can begin to understand some of the feats which they perform every day. Mr. Klein multiplies together numbers of up to six digits by six digits faster in his head than an ordinary man can do by using a desk calculating machine. For example, he wrote down the products of six pairs of three-digit numbers in nine seconds; an experienced calculating-machine operator took a minute to do the same calculations.*

Mr. Klein multiplied

$$1388978361 \times 5645418496 = 7841364129733165056$$

completely in his head, a calculation which involved twenty-five multiplications each of two two-digit numbers and twenty-four

* Kiyoshi Mastuzaki may have been a calculating prodigy, using his abacus as Mr. Klein might use his pipe—to occupy his fingers (see page 6).

additions of four-digit numbers—forty-nine operations in all—in sixty-four seconds. The reader might care to try it himself with pencil and paper. A dozen of us tried it here in Manchester; the times we took varied between six and sixteen minutes, and all our answers were wrong excepting one.

Professor Aitken's students tell many stories of the prodigious ability in mental arithmetic which he demonstrates in his lectures. For example, he is accustomed to ask members of his class to give him at random nine numbers, each of two or three figures, which will form the elements of a 3×3 matrix. He then mentally evaluates the nine co-factors and the determinant, thus obtaining the adjugate and the inverse matrix. He also works out all four roots of a quartic equation with real roots, the coefficients of which have been given to him by his class.

As an example of Professor Aitken's methods we shall describe the operations which he performed while he was mentally evaluating the square root of 567, which he finally checked by comparing his answer with $9\sqrt{7}$. His method is based upon the fact that if a is a first approximation to $\sqrt{n}$ then $\frac{1}{2}(a + n/a)$ is considerably closer, but his calculations are facilitated by his astonishing familiarity with tables of reciprocals.

He noted 24 as a first approximation, and 23·8125 as a second $(23{\cdot}8125 = \frac{1}{2}(24 + 567/24))$. At the same moment he recalled that $1000/42 = 23{\cdot}\dot{8}0952\dot{3}$ the digits of which are close to 23·8125. He performed $567 \times 42 = 23814$ almost before he had thought what he was doing. Averaging 23·809523 . . . and 23·8140 he had as a third approximation 23·81176190476. He recalled simultaneously that $1/84 = 0{\cdot}0117619047619\ldots$ and in extraordinarily less time than it takes to describe it, perhaps in three seconds at most, he registered 23·811762 as the square root he wanted. But now, how many places are right? He noted that the mid-point of 23·8095 and 23·8140 commits an error of deviation of 0·00225 or relatively, 1/10583. Like lightning he squared this and halved it, and reduced his first answer by one part in 224 million, obtaining 23·8117617985, and announcing as a second result 23·81176180 (in fact it is 23·8117617996). "It would be unreasonable," says Professor Aitken, "to ask for anything more accurate. Words cannot describe the speed of association in these matters, and the resources upon which the memory and the calculative faculty draw. The will rises and makes a most powerful imperative; brain and memory obey like an electric switch."

All calculating prodigies acquire, with long experience, an astonishing familiarity with the properties of numbers. For example, Professor Aitken was once asked to multiply 123456789 by 987654321; he immediately remarked to himself that 987654321 is 80000000001/81, thereby converting a tedious sum into a "gift." Asked for the recurring decimal form of 41/67, he multiplied numerator and denominator by 597, obtaining 24477/39999, and writing down immediately $0\cdot\dot{6}1194029850746268656716417910447\dot{7}$.

Even as a schoolboy he was able to astonish his fellows by squaring 57586 in his head in two seconds. He worked it out as follows, using the formula $a^2 = (a - b)(a + b) + b^2$.

$$\begin{aligned} 57586^2 &= 57500 \times 57672 + 86^2 \\ &= 23 \times 144180000 + 86^2 \\ &= 3316147396. \end{aligned}$$

These short cuts, which are an essential part of the repertoire of all mental prodigies, are quite beyond the scope of a machine, which makes much better time by using straightforward methods once the problem has been explained to it; but it is in this sort of way that Mr. Klein performs a type of computation in which he has a most unusual skill. He can express prime numbers of the form $4n + 1$ as the sum of two squares; if they can be expressed as $8n + 1$, in the form $2c^2 + d^2$, etc. For example—

$$\begin{aligned} 5881 &= 75^2 + 16^2 \\ &= 2 \times 54^2 + 7^2 \\ &= 3 \times 32^2 + 53^2 \\ &= 5 \times 9^2 + 74^2 \\ &= 7 \times 24^2 + 43^2 \end{aligned}$$

all of which he did in 100 seconds. Any machine would take a relatively long time to do such a computation, as it would have to work by a tedious process of trial and error. A mathematician would probably take an hour or so to prepare a tape of instructions for a machine which was to handle the general case, a second or so to feed in the particular number which he wanted to investigate, and the machine would take two or three seconds at most to produce the whole series of squares once it had started to work. The point is, of course, that the machine has to be told everything it needs to know for this particular problem, and even when it knows how to proceed it may well take so long for a man to pose a problem to the machine that a human calculator may have done the sum long before his colleague has had time to punch a tape with which to feed the numbers into the machine.

A calculating prodigy draws continually on the accumulated experience of a lifetime's arithmetic and both his "strategy" and his "tactics" are opportunist. Professor Aitken says: "Though these processes take time to describe, they pass in the mind with prodigious speed, though with the ease and relaxation of a good violinist playing a scale passage. Often the mind is so automatic that it anticipates the will.

"The power of numerical memorizing came to me later than verbal, but rapidly improved with mental calculation; and soon all three kinds, verbal, numerical and musical advanced equally. If the number to be scanned had a strong mathematical interest, like π or e or Euler's constant γ, or those almost uncanny numbers like $e^{\pi\sqrt{163}}$ which is 262537412640768743·999999999999250 (incredibly close to a whole number), then I could hardly help absorbing them to very many decimal places.

"The numbers come into view as one needs them, but even to say that they come into 'view' gives a false impression. It is not 'seeing' in the ordinary sense; it is a compound faculty that has never yet been accurately described. The analogy of music will throw light on calculation. The violinist (unless he is momentarily in a difficulty) does not need to visualize the notes on the stave, or the fingering or the bowing; the melody is everything—he is caught up in what he is playing. So it is with the mental calculator; visualizing occurs last of all, and only as required when all else has been done."

The rest of us must be content to marvel.

Mr. William Klein's brother Leo, who died at the hands of the Gestapo during the war, was almost as good a computer as William, and a better mathematician. Dr. Stokvis, of Amsterdam, made a psychological study of the brothers;[(1)] he found that although their performances were very similar, their methods of operation were quite different. For example, Mr. William Klein remembers numbers "audibly"; he mutters to himself as he computes, he can be interrupted by loud noises, and if he ever does make a mistake it is by confusing two numbers which sound alike. Leo, on the other hand, remembered things "visually"; and if he made a mistake it was by confusing digits which look alike. Both brothers were fascinated by numbers from their earliest childhood; William practised arithmetic almost all the time, but Leo hardly ever. Leo studied mathematics at the university, but William read medicine, took a medical degree, and had "walked the hospitals" before he finally decided to earn his living as a computer. Dr. Stokvis investigated the effect of drugs and of hypnosis on Mr. William Klein, and found that neither improved his performance as a computer if he was using methods in which he was experienced and in which he had already achieved an "optimum" performance. Apparently Mr. Klein forgot to go to Dr. Stokvis's lecture so that the public demonstration of his talent which had been arranged had to be indefinitely postponed.

Many other mathematicians whose skill in arithmetic was much less than that of Professor Aitken or Mr. Klein have nevertheless been fascinated by the properties of numbers. Professor Hardy once visited the Indian mathematician Ramanujan, who was lying ill in hospital. To make interesting bed-side conversation Professor Hardy remarked that the number of his taxicab was 1729, which is a multiple of 13, and said he hoped that this was not an ill-omen. "On the contrary," said the sick man, brightening up at once, "1729 is a beautiful number; it is the smallest integer which can be expressed in two different ways as the sum of two cubes." That it can be so expressed is fairly obvious; to prove that it is the smallest such number may occupy the reader in leisure moments for some time, but he may derive, in the process, some idea what mathematicians with a gift for arithmetic talk about in their spare time.

The reader may feel that he is overwhelmed by the possibility of this kind of calculation, but before he decides to take up farming instead of arithmetic let us for one moment consider the mental arithmetic which is sometimes done by a certain Lakeland shepherd. During the course of a day his dog may drive past him a flock of perhaps two thousand sheep. At the end of the day he knows not only *how many* sheep are missing, but *which* sheep are missing. Now even if one assumes for purposes of argument that a man can learn to tell the difference between one sheep and another, one must admit that even a shepherd requires and can exploit a skill in mental arithmetic which few of us could ever hope to achieve.

We have seen how skilfully some fortunate men can do arithmetic, and it is impossible for us to evade any longer a discussion of the interesting but not perhaps very important question: "Can these new machines think, and if not, will it ever be possible to build machines which do?" The question has already been widely discussed in the press—for example, in the correspondence columns of *The Times* last year. It appeared to a casual reader that, as so often happens in controversies of this type, the argument was really about the precise meaning of the words themselves rather than about the machines.

In the thirteenth century a Spanish mystic, Raimor Lull, built a machine with which he claimed to solve logical problems. He called it his *Ars Magna* and with it he was firmly convinced that every truth in Christian revelation could be proved. He used it for instance to investigate the Attributes of God and he hoped that it would help him to convert Moslems to Christianity. Unfortunately his fellow Christians were sceptical and the Moslems stoned him to death in 1315. Francis Bacon called the Ars Magna a *Methodus Imposturae*, but it had a great reputation in its day and it may have suggested to Swift the machine which Gulliver saw in use in Laputa. A big square frame contained hundreds of small cubes strung together by wires. On each face of every cube was written a word in the Laputan language. A Professor and his pupils turned cranks that rotated the cubes and whenever the words which were visible on the cube faces made sense, the scholars copied the phrases down. "By this contrivance," said Swift, "the most ignorant person may write books in Philosophy, Poetry, Politics, Law, Mathematics and Theology without the least assistance from genius or study."

Sir Arthur Eddington popularized a very similar scheme when he suggested that, given long enough, an army of monkeys could type out all the books in the British Museum. This method of operation, which is commonly known as "the fifty million monkeys technique," appears to a bystander to be more widely used than one might suspect as an experimental method (always, of course, in other people's laboratories). It has the disadvantage that progress made is proportional only to the square root of the time spent, since it is an entirely random process. Chapter 15 contains a detailed analysis of the possible application of random numbers to logical reasoning, and shows, astonishing though it may appear at first sight, that they may under some conditions have a perfectly serious and genuine application. We must, however, turn to less exotic and more familiar subjects.

Anyone who has watched an automatic pilot controlling a giant air liner, and bringing it almost all the way across the Atlantic with no help from the human beings on board, may have wondered whether it is not reasonable to imagine that "George" is thinking for himself. Instruments of this type have "grown up" gradually stage by stage, they perform "calculations" of a fairly simple type which can be done almost instinctively after long practice by the human beings whom they so ably assist, and, perhaps most important of all, these machines are quite unoriginal. Each type is capable of only one sort of computation—for example, "George" cannot drive a car. Few people have seriously insisted that an automatic pilot "thinks" for itself in the sense that its designer thought when he was making it.

The problem has to be reconsidered when one is confronted with the achievements of a full-scale automatic digital computer, which can undertake calculations of enormous scope and variety, and which appears in many instances to be solving problems which no human being could encompass in a lifetime. Can we be quite certain that these machines do not "think," or that their successors in a few years' time perhaps will not be able to think, in a manner indistinguishable from the mental processes of human beings?

Lady Lovelace realized that a machine would be able to guide itself through a calculation of great complexity by following out in detail a routine which had been prescribed for it, and that it could reach decisions by following a mathematical version of the game of "Twenty Questions." We have already quoted her dictum, "The Analytical Engine has no pretensions to *originate* anything. It *can do whatever we know how to order it* to perform" (page 398). This statement is quoted by Hartree,[(2)] who adds—

"This does not imply that it may not be possible to construct electronic equipment which will 'think for itself,' or in which, in biological terms, one could set up a conditioned reflex, which would serve as a basis for 'learning.' Whether this is possible in principle or not is a stimulating and exciting question, suggested by some of these recent developments. But it did not seem that the machines constructed or projected at the time had this property."

We shall return to this aspect of the problem of "learning" in a moment, but let us first consider the implication of the ability, which the machine already possesses, of following out a routine which has been given to it, or, if one can use the word without begging the question, which the machine has been "taught" by its programmer. Any human computer who was following the same routine would be most offended if he were told that he was not actively thinking while he computed.

It may be possible to programme the machine to bid and to play a good game of bridge. If so, this achievement might well mark one more step on the road along which mankind has struggled to relief from profitless tedium and intellectual strain. The machine would play by following prescribed numerical criteria but, after all, so does the ordinary bridge player, although he is usually far less skilled than the machine would be since his capacity for mental arithmetic is limited and he does not always remember the fall of the cards. The machine would be unable to indulge in the subtler forms of cheating which can add so much to family bridge, but apart from that it might play as well as an average man. How then can we distinguish between the processes which go on in the brain of the bridge player and those which go on in the machine? It does not seem safe to say dogmatically that one of them is thinking and the other is not if both of them achieve similar results with similar information.

Let us return for a moment to our discussion of the machine's treatment of Babbage's problem (see page 75). We said then that the machine would never deduce by itself that there is no integer whose square ends in the number seven, whereas a human being would know this. It is fair to inquire if we were not a little over-hasty in making this assertion so confidently. In the first place, it is undoubtedly true to say that there are many human beings who would never be able to generalize from their experience of forming tables of squares and so deduce that no such square exists, but one cannot necessarily, for this reason, deny them the rank of "thinkers." Furthermore, although it is true to say that, if they were told to look for generalizations of this kind, they would probably spot them, the machine would do so too if the problem were propounded to it properly. The machine is, in fact, capable of learning from experience, of discovering, for example, the types of squares which exist, and rejecting any problem which requires it to find one of a type which it has previously discovered to be non-existent. This is the simplest possible case of learning by experience; it is the kind of thing human beings do and pride themselves on, as evidence of their power of conscious thought.

The machine might in much the same way be made to improve its skill at chess. It could play against itself and find out which of several rules of play was best, finally adopting that which made it win most often. Oettinger[3] has considered this problem in some detail, and has written a programme for the E.D.S.A.C. which makes it learn a few simple lessons from experience. He shows, for example, that if it "plays at shopping" it will learn by trial which goods are available at which shop and order things from the right place. It can, moreover, cope with the possibility of shops running out of some items.

Turing[4] has analysed the problem from another point of view, and suggests that in a few years time it may be possible to make a machine which will answer almost any question asked of it in the same sort of way as a human being would do, so much so, in fact, that it would be difficult for anyone who posed the questions simultaneously to a machine and to a human being to discover from the answers which was which (or who was who). When he is playing this "imitation game" the inquisitor must be able to see neither of his victims and all communications must take place in code. Turing assumes, of course, that the machine would be trying to "pass itself off" as human and that it would for example type out a statement that it has blue eyes and blond hair and likes strawberries and cream, if asked the appropriate questions. The machine might make a few "mistakes" in arithmetic from time to time so that the inquisitor would not be able to identify it because of its infallibility. Turing thinks that in ten or twenty years' time it should be possible to build a machine which would in this "imitation game" leave more than half its inquisitors in doubt of its "humanity" for five minutes or so.

In his Lister Oration for 1949, Professor Jefferson said:

"Not until a machine can write a sonnet or compose a concerto because of thoughts and emotions felt, and not by the chance fall of symbols, could we agree that machine equals brain—that is, not only write it but know that it had written it. No mechanism could feel (and not merely artificially signal, an easy contrivance) pleasure at its successes, grief when its valves fuse, be warmed by flattery, be made miserable by its mistakes, be charmed by sex, be angry or depressed when it cannot get what it wants."

This statement appears at first sight to be incontrovertible, but what happens if we invert it and apply it to human beings? As Turing* points out, in its extreme form the argument implies that the only way in which one can be sure that the machine thinks is to *be* the machine, and to feel oneself thinking, but logically it is equally true to say that any proof that a human being other than

* Loc. cit.

oneself is thinking should depend on a similar transmigration of souls. We usually make the assumption that other human beings do think, if only because it makes for greater simplicity all round (even if we restrict our concession to *some* human beings). It is rather illogical to refuse a similar courtesy to the machine because it can only think about a limited range of subjects, and it seems unfair to refuse it the title of "thinker" merely because we imagine it to be incapable of appreciating strawberries and cream as we may have taught it to say that it does.

Many people have asked with Professor Jefferson whether the machine would be capable of composing a sonnet. It seems certain that it would be possible to programme a machine in such a way that it would produce an assemblage of words which satisfied the rhythmic requirements of sonnet construction and which rhymed in the proper way, but we do not know at this stage how we could ensure that the production would be grammatically and syntactically correct. It may be possible to do this before too long, but we have no idea how we could ever ensure that the product would be aesthetically satisfactory to a human being; in other words, the sonnet would be such as to appeal only to another machine.

To build a machine that would write something like *Ulysses* in sonnet form might tax the ability of several generations of Babbages, and as Watson-Watt(5) put it, the life-long efforts of a potential Shakespeare might be devoted to the construction of a mechanized Martin Tupper. It is easy, however, to make this point too strongly. We know of no potential Shakespeare who is devoting his efforts to digital computers at the moment, nor do we think it likely that such a man would do so if the opportunity were given to him. He would make far more money in Hollywood. (We must confess that we cannot be so certain about potential Bacons.) We must point out furthermore that the efforts of computer engineers have already produced a mechanized Briggs (who spent his lifetime computing logarithms) and a mechanized Barlow (whose famous Tables were his life's work) but no one has ever conceived of a mechanized Napier (for he *invented* logarithms). It will be better to devote ourselves to the development of machines which can handle problems of a type more suited to their abilities, and to remember that calculating prodigies are less important than original thinkers.

It already appears that over a limited range of types of intellectual activity the differences between the operations of the machines and those of a human being are of degree rather than of kind, so let us for a moment contemplate the structure rather than the functions of the human brain.* The average brain contains about ten thousand million individual cells, or approximately five times as many cells as there are human beings in the world. Each cell is connected, directly or indirectly, to every other, and small electrical impulses flow through the nerve fibre connecting them at a speed of between fifty and two hundred miles an hour. A modern digital computer may perhaps use three or four thousand valves. It is doubtful if all the valves that have been made since Lee de Forrest's time amount to a fifth of the number of cells in the brain of an idiot child. If they could be assembled together in one place they would require all the buildings in Whitehall to house them, several Battersea power stations to drive them and the Thames to cool them. Even if they were of the best modern type about 1,000 would fail and need replacing every minute. Moreover the problem of interconnecting them would daunt the imagination of the most stout-hearted engineer.

Machines are in a very early stage of development; their structure, by comparison with that of a human brain, is exceedingly simple; nevertheless they can perform operations which tax the abilities of a thinking human being to the utmost. It already appears that the difference in complexity between man and machine is greater than the difference in ability. The structure of the brain of an ordinary earthworm is as complicated as that of a modern computer. The computer achieves its success because the whole of its "abilities," limited though they are, are directed by a mathematician to some particular end. This fact of course is the measure of the skill and ability of the programmer, and it is a pity not to reverse the argument and speculate for a moment on the potentialities of a human brain if it were efficiently organized by some "programmer" or other. It is well known that, when he has been hypnotized, a man can often recall in detail all kinds of incidents that completely escape him when he is in his normal conscious state. The extent of this power of recall is very surprising. For example, a bricklayer carefully inspects each brick as he lays it, but he makes no attempt to remember the tiny cracks and other marks which can be used to distinguish one brick from another. In a

* The relation between "mind" and "brain" has long been studied by philosophers and psychologists and the authorities express their opinions with confidence. Professor Wilder Penfield wrote: "Everyone knows that the mind of man depends on the action of the brain," and Professor Aveling, even more emphatically: "The view that the mind is a product of matter need hardly be taken seriously."

This is a subject about which we know nothing whatever, and we can only urge the reader to form his own opinion.

recent experiment half a dozen bricklayers were hypnotized and each was asked such questions as, "What shape was the crack in the fifteenth brick on the fourth row above the damp course in such and such a house?"—a brick which the man hadn't even seen for several years. In an astonishingly large number of cases the man could remember. It is easy enough to store information in our own minds, in a library, in a filing system or in a computing machine, but it always seems to be exasperatingly difficult to get it out again. It is most frustrating to discover that one really possesses an infallible memory, but that one can only explore it properly if one submits to hypnosis by someone else, who doesn't really know what is inside it. It is difficult for a hypnotist to improve the reasoning powers of his patient, though every schoolmaster claims to do so, but there does seem unfortunately to be a limit to the improvement to be expected in a man's mental powers, however well his thoughts are programmed.

It is probable that the next advance in the field of computer design will involve a great increase in the storage capacity of the machines. We have already stated that few modern computers have a store as big as a tenth of the size that Babbage had planned. We mentioned the fact that such huge stores might be made by using magnetic tape, made familiar to all from its use in sound recording, and it is worth while diverging for a moment from our theme to point out the connexion between the storage of digits and the "storage" of music. It is possible to specify the position of the cone of a loudspeaker by using a series of groups of binary digits in such a way that its movement so described produces a reasonable facsimile of a piece of music. It is necessary to specify the position often enough to ensure that the highest required frequency is radiated (this determines the number of groups of digits) and with an accuracy which (in principle) is fixed by the ratio of the intensity of the music to that of the unwanted background (this quantity determines the number of digits in each group). The formula which gives the number of binary digits required per second is—

$$F \times \log_2[1 + (S/N)^2]$$

where F is the frequency band, and S/N the signal to noise ratio.

For good sound reproduction F is about 10,000 cycles and S/N about 1,000, so the number of binary digits required per second is about 200,000*. A second's music is recorded on about twenty inches

* In fact because the ear is very tolerant of certain forms of distortion, this number is an overestimate as we shall see in a moment.

of wire or tape. In other words, the amount of information which it is necessary to record to produce a reasonable facsimile of a symphony lasting for an hour is equivalent to something like 700 million binary digits or about 1,000 times the capacity of the Manchester machine. It is not possible, as we shall see, to fill a tape so tidily with digits as with music, but it is obvious that the storage of information corresponding to two thousand million binary digits, for example, is quite feasible. Is so happens that the contents of the Encyclopedia Britannica can be expressed in binary form in about this number of digits.* It would be possible to provide a machine with a storage system of much greater size than anything which has so far been achieved. The tape itself would cost only a few hundred pounds; if the same amount of information were stored on teletype punched tape it would cost three or four times as much and occupy a hundred times as much space. If it were punched on cards the cards themselves could cost several thousand pounds and fill an ordinary room.

The problem of obtaining access to this information and of classifying it in such a manner as to be able to make use of it is of course extremely difficult and has not yet been faced, but it is important to appreciate the implications of the fact that it is already possible to store the contents of an encyclopedia on wire or tape in binary form, and that the store need not be much bigger than the original book.

In the first place, a machine which had a memory as large as this would be a formidable opponent in any "imitation game" because it could learn an appreciable portion of the sum total of human knowledge, in so far as this has been reduced to book form, and since its memory would be permanent and all parts would be freely accessible it might be capable of a great variety of "mental" processes not so far considered to be suitable for a machine. We shall consider this point again in a moment.

First of all, however, let us take up another point which emerges very clearly from this rather simple analysis of the information content of a gramophone record. This is the astonishing rate at which a human being can absorb "information" through his senses. A trained musician can appreciate the overall effect of a gramophone record and simultaneously detect very small imperfections in it.

It by no means follows, however, that this "information" in all

* A group of 6 binary digits can define any one of 64 characters, for example all letters of the alphabet, small and capital, and the numbers from 0 to 9; we have not allowed in our estimate for abbreviations and elimination of redundancies. The Braille alphabet is written in a six-digit code.

its detail ever reaches the level of consciousness. We have already explained that a listener usually ignores a few "clicks" on a gramophone record; it is possible to reduce the "information" in a recording channel a hundredfold before the listener fails to recognize a simple tune. The truth is, apparently, that the senses make great use of "redundancy" as a method of assuring that the final result which reaches the brain is correct, in much the same way as (but to a much greater extent than) a digital computer.[6] The U.N.I.V.A.C. consists of two complete machines, one of which constantly checks the operations of the other. There are scores of such channels in the human brain which filter the incoming information before presenting it to the consciousness, thereby avoiding error at the expense of the loss of the greater part of the incoming information to which the senses are exposed. Moreover, some part of the analysis seems to take place in the unconscious mind; a good musician can fill in the rest if the record gives him a hint of the music, and he may be quite unaware of the extent to which his mind is supplementing his senses. A musically illiterate engineer is far more likely to detect the onset of distortion in the reproducing system. Most people can understand snatches of conversation in their own language, but lose the thread of a conversation in a foreign language unless they hear everything very distinctly.*

After the senses have dealt with the incoming information the brain proceeds to interpret it in the light of long experience. It is quite true that an acute and well-trained ear can detect the degradation in musical quality if the reproducing channel is capable (in theory) of handling much less than 200,000 digits per second, but this phenomenal amount of information is really necessary only because the gramophone has no musical intelligence whatever; it could with equal fidelity handle any sort of noise; music is subject to certain rules. Most European music is written in subdivisions no finer than semitones, almost all within a compass of seven octaves, or eighty-four semitones in all. A note can therefore be expressed by seven binary digits. Assume as a reasonable average one bar per second, eight chords to the bar, and four notes per chord, if there are more they can certainly be constructed by the mechanical rules of harmony. This gives $7 \times 8 \times 4 = 224$ binary digits per second for the specification of piano music. There is no

* It is presumably for this reason that so many Englishmen and Americans shout at French porters, on the assumption that if it is loud enough anyone can understand English. We once heard an English officer at Ostend shouting in Hindustani, but he was producing no visible results.

need to count changes of tempo or loudness as they only occur relatively infrequently. In fact we have already a factor of perhaps two too much, as a musician could guess half the information from the rules of harmony, so that 200 digits per second is more than enough for piano music, and may be enough for orchestral music. Suppose that a conductor such as Toscanini knows by heart a couple of hundred hours of music; on these assumptions his whole musical memory contains about 150,000,000 binary digits, or as many as the gramophone handles in twelve minutes. Toscanini will notice if the oboe is a quarter-tone out, but even theoretically it will take him anything up to a tenth of a second to do so; it follows that he cannot detect many such errors per second, and if several similar errors occur at once he will probably give up in despair. The point is, once again, that one learns to ignore everything except the one thing on which one is concentrating at the moment.

This question is so interesting that we must study it in a little more detail. An ordinary telephone line can be used to transmit words in the standard teleprinter code, and it can be shown that if a line is used in this way it will handle about two hundred times as much information as can be passed through it if it is being used to transmit conversation. This estimate is made on the assumption that the tone of voice of the speaker and what one may call the "emotional content" of the message are unimportant. The redundancy in speech is in fact so great that the most hideous distortion of the wave form does not make it unintelligible. For example, the waves can be "clipped" or "limited," and the resultant series of square waves differentiated. This operation completely changes the shape and the frequency spectrum of the message, but although one would be hard put to to discover any point of similarity between the input and the output, the signal remains quite comprehensible although it sounds like a talking buzz-saw. All that has been preserved are the points at which the wave crosses the axis, yet these convey the meaning. The transmission of "start-stop" signals which modulate half a dozen oscillators will also suffice to transmit intelligence. In this case the distortions are quite different, but still inadequate to destroy the meaning. We are now beginning to understand how fortunate the pioneer telephone companies were, it didn't matter very much how bad their system was, it was still bound to work.

There is a considerable amount of redundancy in the written language too. One can usually guess a word in its proper context even if a few letters are missing, and one can usually understand a message if a few words have been lost. It is this sort of thing which makes crossword puzzles possible. The idea of redundancy is very widely exploited; for example, the navigating officer of a ship or an aircraft never depends entirely on one method of position fixing, but he ensures the safety of his vessel by comparing the results of two or three quite different methods of navigation. Commercial accountancy makes extensive use of redundancy as a method of assuring accuracy. All items are entered into the books of account twice, and the totals are then determined by two different addition sums in which the numbers are usually written in a different order. This process, which is known as Book-keeping by Double Entry, has acquired a status and even a Mystique of its own. The fact that the books balance, which simply means that the mechanical operations of accountancy have been performed without error, is held to be of major importance in itself, and it is presumably for this reason that we have become accustomed to the idea that a company should express its balance sheet to eight or nine significant figures although in fact the total value of its assets is probably in doubt by at least twenty per cent.

It is important to realize that, in all the different examples we have considered, the redundant information conveys the same message in several different ways. This is far better than mere repetition of the same sort of information. It is for this reason that it is so much better to "difference" a set of tables than it is to repeat the computation of each entry; on a much humbler plane, this is why it is so useful to "cast out nines." A mathematician will feel much more confident if a formula can be verified in a particular case by a simple argument than he will be if he repeats his analysis and gets the same answer. Unfortunately, in a computing machine it is very difficult to devise many simple uses of the principle of redundancy. Much of the arid beauty of mathematics stems from its unambiguous and non-redundant symbolism, but it is at least arguable that a type of mathematics which had something comparable to the redundancy of English prose would be much more easy to handle. Good intentions would take a man much further and it is certain that the use of such a system would avoid many of the difficulties which we have described in Chapters 4 and 5. Even more useful would be a method of incorporating the redundancy of functional parts by which the brain is able to carry on apparently unharmed if quite a large part of it has been damaged—the healing process is quite automatic. The best a machine can do is to say where the pain is (see page 86) which is something a man cannot do if the fault is within his own brain.

We can make effective use of redundancy in the storage of information. When we record music we can pack the equivalent of ten thousand digits on an inch of magnetic tape; but when we do this we are quite content if we can be reasonably sure of most of the information we have recorded. We have to be much more conservative if we are recording numbers. We can, for example, write about a hundred digits to the inch, and find that the recording process then introduces an error once in every ten thousand digits. By writing everything in triplicate on tracks which are far enough apart to ensure that a single blemish on the tape cannot affect all three tracks one can reduce the chance of errors occurring simultaneously in two tracks, and thereby introducing a mistake in the final result, to about one in a hundred million. This is tolerable if one can then apply a check such as "casting out nines."

It is remarkable that a phenomenon which one might at first think to be evidence of the relative inefficiency of our senses, which never fully load the information channels which they use, turns out in fact to be of fundamental importance to us all in avoiding errors; and that systems which exploit redundancy in much the same way will have to be introduced into our great computing machines, in spite of the complexity of the special equipment which will be needed.

The human eye can absorb information even faster than the ear —the information passes between eye and brain on more than a million separate fibres. A good television channel will handle the equivalent of more than 20,000,000 binary digits per second, but it is capable of transmitting a meaningless pattern of dots. It has been shown that only about five per cent of the channel is really needed to transmit an ordinary picture, owing to the fact that most of it consists of large featureless areas,* and no one seems yet to have estimated how much of this five per cent is concerned with redundant transmission of the static background and how much with movement. The eye has a remarkable ability to select the significant item and to ignore the rest; think for a moment what must be

* This means in practice that the whole of the frequency spectrum allotted to a television channel is not filled with useful information. In fact the radiated energy is concentrated into a series of narrow bands which have gaps between them. It has been found possible to exploit this fact, and to radiate the extra information which is needed to provide "colour" in the gaps which occur in the spectrum of an ordinary black and white picture.

involved in looking at a tree, ignoring the detail of leaves and branches, and concentrating on a small bird.

The extent of visual memory is immense and almost impossible to estimate. Half an hour ago the writer was walking down a busy street and recognized from behind a friend whom he hadn't seen for twenty years, a man who had lost all his hair and gained sixty pounds in the meantime. What does this type of memory involve?

It has recently been shown that blind people whose sight is first given to them when they are grown up have to undergo a prolonged and difficult period of training before they can make use of their eyes in a way which most people regard as instinctive and automatic; for example, it may take such a person years to learn to appreciate the fact that a yellow square, the sides of which are horizontal and vertical, is the same shape as a red square, half its size and inclined at an angle. In other words, for an individual to learn to appreciate the *gestalt* or squareness of a square in various orientations is very difficult and such an ability is acquired only (if at all) after years of experience, which usually take place before the period of conscious memory, at the same time as the child is learning to understand its mother tongue.

Ehrenfels called attention to man's ability to appreciate certain phenomena which are related to *sets* of stimuli. For example, such qualities as "slenderness," "regularity," "roundness," "angularity," or the characteristic appearance of a circle, a triangle, or other geometric shapes. In German the word *Gestalt* is often used as a synonym for form or shape and Ehrenfels used the name *Gestaltqualitäten* for all of them. Animals with much simpler brain structures than man have this sense of *Gestalt;* as Adrian has pointed out, even a rat can be trained to recognize a circle or a triangle or the more abstract "triangularity."

It appears that an appreciation of the meaning or the *gestalt* of sounds or shapes is often derived by the brain, consciously or unconsciously, by the analysis of redundant information. The processes which are involved in the analysis are beyond our present understanding, but we know that they often depend upon reference to enormous quantities of stored information which, however imperfectly remembered, may represent the essence of a lifetime's experience. We no more understand how we recognize a square when we see it than we know how we recognize the word "square" when we hear it on a bad telephone circuit. Now, as we have seen, machines have only the most rudimentary ability (with which they have been endowed with great difficulty) to exploit the simplest type of redundancy to ensure the accuracy of their calculations; furthermore their memories are of limited capacity, and it is certain that no one can programme a machine to solve a problem he doesn't understand himself, so it is not therefore surprising that it is precisely in this mysterious ability to take an overall view which human beings find so difficult to learn that the machine at the moment seems to be weak by comparison with any human brain.

No one has yet devised a method by which a machine can recognize the "threeness" of a figure 3, so that it can recognize it in any handwriting,* and it would probably be difficult to make it copy instructions from dictation. Nevertheless the success of Bell Telephone Laboratories in producing "visible-speech" patterns by displaying the Fourier spectrum of speech on a cathode ray tube in such a manner as to allow a deaf man to interpret it, and the production by the British Post Office of the Vocoder which compresses the total spectrum required for intelligible speech transmission, make it clear that a machine could be taught to listen intelligently to spoken numbers if it is necessary to do so. In particular it should not be hard to make a machine recognize perhaps a couple of dozen different sounds, which might for convenience include most of the numbers from 0 to 9, so that it could understand the relatively few symbols that are needed to encode numbers and instructions. It would not be asking too much of human beings that they should learn to speak these sounds distinctly and use them when they were talking to the machine (for example, when they were preparing an inventory) so that it could understand them whoever spoke them.

Aesthetic judgments, which usually involve the consideration of the overall effects of sounds or shapes, will be outside the proper province of computing machines for many years to come. It seems most improbable that a machine will ever be able to give an answer to a general question of the type: "Is this picture likely to have been painted by Vermeer, or could van Meegeren have done it?" It will be recalled that this question was answered confidently (though incorrectly) by the art critics over a period of several years. The machine would of course be able to remember the date of Vermeer's birth and the titles of all his known paintings, which is more than some critics could do.

We often find that people can size up a most complicated

* A machine has recently been built to "read" Braille.

situation almost instantaneously, far too quickly for any logical analysis to have been possible, and that their first appreciation is more reliable than the results of conscious thought. This is in fact the argument for the reality of that rarest of human abilities, usually described as "common sense." It has been said that "we expect more from our statesmen than that they should arrive at logical conclusions—all that matters is to be right, and unless instinct rules and reason serves right judgments will rarely be arrived at."* The entire judicial system of this country is based upon the supposition that ordinary men can make decisions in this way. For consider: a jury consists of men who are ignorant of law, unaccustomed to hearing and interpreting evidence, lacking the experience and mental agility of professional lawyers, and usually destitute of scientific training. All reason suggests, therefore, that they must be quite unable in a complicated case to weigh the evidence properly and arrive at a reasonable verdict. Nevertheless, said Lord Chancellor Halsbury (who sat on the Woolsack longer than any other man), "As a rule, juries are in my opinion more generally right than judges." The great Lord Mansfield advised an inexperienced judge to announce his decisions but never to give his reasons, for, he said, "Your judgments will probably be right, but your reasons will certainly be wrong."†

Since we don't know ourselves how we do this sort of thing and few of us ever learn to do it well, and since decisions of this type do in fact seem to demand the subconscious evaluation of a situation with the background of a lifetime's experience, there is little wonder that the problem of giving such an ability to a machine has baffled everyone since Babbage who was well aware of the problem and remarked that men who can effectively base their decisions on a proper appreciation of a complicated situation are very rare. It is, he said, the quality which is most needed by a successful general, and he added that the Duke of Wellington, whom he knew well, possessed this ability to an outstanding degree. The most difficult of a general's problems are due to never-ending changes in his data as the battle progresses. Very different is the problem of a composer who has to appreciate the *gestalt* of a whole piece of music. Mozart

* OLIVER. *Ordeal by Battle.*

† The judge did very well as long as he followed this advice, but some years later he was accused of gross incompetence, and Mansfield had to sit on a commission which was to decide whether to remove him from office. It turned out that his judgments had in fact been uniformly correct, but his continued success had made him over-confident, he had taken to announcing his reasons, and they were palpably absurd.

has told us that he conceived the whole of a piano concerto at once before he wrote any of it down.

Babbage's own judgment often seems to have been unreliable when he dealt with practical matters, and this letter, which he wrote to Lord Tennyson, suggests that even his aesthetic opinions had a quantitative bias—

> Sir,
>
> In your otherwise beautiful poem ("The Vision of Sin") there is a verse which reads—
>
> Every moment dies a man,
> Every moment one is born.
>
> It must be manifest that if this were true, the population of the world would be at a standstill. In truth the rate of birth is slightly in excess of that of death. I would suggest that in the next edition of your poem you have it read—
>
> Every moment dies a man,
> Every moment $1\frac{1}{16}$ is born.
>
> Strictly speaking, this is not correct, the actual figure is so long that I cannot get it into a line, but I believe that the figure $1\frac{1}{16}$ will be sufficiently accurate for poetry.
>
> I am, Sir, yours, etc.

A machine might well have inspired this, for it is reasonable and logical, apparently* serious—and quite irrelevant.

Yet we must not assume too hastily that a machine cannot cope with a situation merely because it is complicated. Once the complexities become too great for one man to handle, the difficulties of communication between man and man may be such that the machine begins to show up favourably by comparison. This is why we feel that it ought to pay to mechanize a big office. Let us consider for a moment a more straightforward problem: all large airlines have great difficulty in handling their seat reservations. They lose money unless two-thirds of all seats are sold, so they cannot use the same system as a bus line; and they find themselves saddled with a huge organization which may cost more than the petrol which the aeroplanes use (in some cases nearly twice as much). Trunk telephones connect all the booking offices in the country to a central office in which scores of girls look at flight information written on a series of blackboards. There are so many blackboards,

* Apparently! The reader will probably have decided long ago that what Babbage really needed was a good public relations officer.

and the girls have to sit so far away (there are so many girls too) that they have to use telescopes to read the blackboards and the system tends to be limited by the power of available telescopes. One cannot but feel that any attempt to improve such a system in its present form holds about as much promise for the airlines as would a long-term research programme to improve the thermodynamical efficiency of horses. It is clearly a job for a computer. A memory of a few million digits, some computing circuits and a series of teleprinter lines should make it possible for any booking office to reserve space on any flight, or to offer alternative space if need be. A machine which solves at least part of this problem is now in use at La Guardia Airport; doubtless others will be built before long.

It is pertinent to ask ourselves what, if anything, distinguishes the processes which take place in these machines from some which go on in the brains of human beings. We shall give the game away completely if we try to restrict the use of the verb "to think" to those mental operations which we don't understand ourselves, which we cannot control, which take place subconsciously and which we strongly suspect of being fundamentally illogical into the bargain. One cannot dismiss the question by saying that the machine is merely a "complicated abacus"—it is the very complexity which poses the problem. "The whole is not the sum of parts; its nature lies in its constitution more than in its parts."* To ask "How complicated must a machine be before this question arises?" is as pointless as to ask "How many hairs can a man lose before he becomes bald?"

Three hundred years ago, Thomas Hobbes considered the philosophical implications of Pascal's invention of a simple calculating machine. The mental effort which its development cost Pascal was so great that his health never recovered from the resulting breakdown. His contemporaries thought that his machine was much less important than his mathematics, his philosophy and his theology, but posterity is even more indebted to him for his invention of the ordinary wheelbarrow. Pascal had written "La machine arithmetique fait des effets qui approchent plus de la pensée que tout ce que font des animaux." Hobbes commented that "Brass and iron have been invested with the functions of brain, and instructed to perform some of the most difficult operations of mind. . . . In what manner so ever there is place for addition and subtraction, there also is place for reason, and where these have no place, there

* General Smuts. *Address to the British Association*, 1931.

reason has nothing at all to do; for reason is nothing but reckoning (that is the adding and subtracting) of the consequences. . . . When a man reasoneth, he does nothing else but conceive a sum total from addition of the partials. . . ." It is difficult not to agree with him, although we have to admit that many of these mental "additions" must take place in the unconscious mind.

Before we conclude this discussion of the current and potential applications of digital computers, it is worth considering for a moment whether machines of this type can assist research work in fields other than mathematics. In particular, is it possible to exploit the fact that a few hundred thousand feet of cine-film would contain a significant fraction of all the "information" which has ever been reduced to print?

Vannevar Bush (who, during the war was scientific adviser to President Roosevelt) has recently concerned himself with the problem of storing and correlating the information which is constantly being prepared and published in books all over the world. He points out that, on an average, the number of books in a large library such as the Library of Congress doubles every sixteen years.[7] Most libraries are already full and it is clear that in about another hundred years' time there will be no buildings in the world large enough if they continue as at present to store all the books which are offered to them. The obvious proposal has been made that books should be stored in microfilm form, but this is only a very partial answer to the fundamental problem, which is how the information contained in them can be made available to people who are engaged in research work and wish to familiarize themselves with the work already done. It is true even now to say that it is often much quicker to repeat a simple piece of research work than to look through the literature to find out if it has been done before—the system of disseminating information seems likely within a foreseeable time to break down under its own weight.

Mendel's concept of the law of genetics was lost to the world for a generation because his publication did not reach the few who were capable of grasping and extending it; and this sort of catastrophe is undoubtedly being repeated all about us, as truly significant attainments become lost in the mass of the inconsequential.

"The difficulty seems to be, not so much that we publish unduly in view of the extent and variety of present-day interests, but rather that publication has been extended far beyond our present ability to make real use of the record. The summation of human experience is being expanded at a prodigious rate, and the means we use for threading through the consequent maze to the momentarily important item is the same as was used in the days of square-rigged ships."[7]

Bush proposed that all incoming literature should be classified and docketed, and that the equivalents of the card indexes should be stored in such a way that a research worker would be able, by dialling a number or two, to obtain access to all the data which existed on the subject in which he is interested. The greater part of the literature which any normal individual would want could, if it were photographed on microfilm, be contained in a single room. It may be possible to solve the problem of storing information in this way and, if the major difficulty of classifying the information adequately can be solved with a machine of the type which Bush proposes, it will assist research workers by giving them access to information very quickly. Furthermore, by drawing their attention to similarities between topics which have hitherto been regarded as totally disconnected, such a machine may perform one of those operations which is now thought of as peculiarly that of the original thinker. Nevertheless, this argument can be pushed too far, as it would be most unlikely for example that in any index an apple and the moon would both have been included under the general heading of "falling" before the idea had occurred to Newton, and it is improbable that any machine would produce its best "ideas" about any subject when it wasn't "thinking" about it, or even more important, that it would recognize them for what they were if it did.

Bush discussed the problems which would confront a man who was investigating the developments of the composite bow which was used by the cavalry of Ghenghis Khan and the effect of its introduction on world history. The would-be author would need to know something about the mechanical properties of the materials of which the bow is made, the aerodynamics of arrow flight, and the history of archery, as well as the campaigns of the Mongols, and no one individual could ever be expected to have an acquaintance with such a wide variety of topics. It is moreover far from clear that he would have sufficient familiarity with the index of published literature to know which buttons to press. All one can say is that the production and development of machines of this kind would be of great assistance to human thought, that they might mechanize many of the more humdrum operations involved in thinking, and might even be of importance to original thinkers in the development of their own ideas by making it possible for them to do things they would never achieve without assistance. One would not normally describe such an operation as one of "thought" in the accepted sense of the word, but it is not easy to define the word in such a way as to exclude these operations if one imagines that they may be performed equally by a machine or by an assistant librarian.

At this stage, the connexion between the Vannevar Bush machine and the ordinary digital computer is not particularly obvious, but we have mentioned it firstly because it is one of the outstanding proposals for mechanizing processes of storing and using information; secondly because the process of selection will inevitably be performed according to some numerical criteria; and thirdly because it is possible that a combination of a digital computer with a memory of the type proposed by Bush is the next step in the development of computing machinery. We have already given an account of the possible application of the machine to weather forecasting. The machine might solve the equations for the movements of the air, or if it had a large enough memory it might recognize a given configuration of atmospheric conditions and base its forecast on its recollections of what this configuration led to the last time it occurred.

It is interesting to speculate on the possibility of using such a machine to help a lawyer in his judgment of a case which was presented to him in such a way as to allow him to decide it entirely by a logical interpretation of statutes and by reference to a well-indexed library of leading cases. If the findings of the machine seemed to lack something of the spontaneity and variety of current legal verdicts, the careful and judicious use of the random number generator might help to mitigate the harshness of pure logic.* It has been argued[8] for example that a decision of the House of Lords might be likened to an Act of God, being something unexpected and unpredictable which no "reasonable man" would expect (unfortunately the argument was rejected by the court). Of course, in England "the law has never recognized the existence of a reasonable woman, whose existence must therefore be regarded as impossible." (Lord Chief Justice Hewart.)[8]

What then are we to conclude from this discussion? It is hard to deny that many of the activities of a digital computer are such as would normally be regarded as evidence of conscious thought were they to be undertaken by a human being. If a distinction is to be made it will require a redefinition of the verb "to think" which has

* But see Chapter 15.

so far not been faced by the lexicographers, and it will be hard to exclude the performance of the machine and admit that the ordinary man ever thinks at all. We have no doubt that the metaphysicians, who understand the manipulation of words much better than that of machines, will find a way out of the impasse in the end, even if they have to use a machine to do it. It is certain that in a few years time they will, like the rest of us, be using machines to help them in all kinds of ways in their everyday lives.

If it is true to say that a machine can do the work of several hundred ordinary men, it is equally true to say that no machine is ever likely to undertake the work of those few extraordinary men whose dreams and whose efforts are responsible for the growth and the flowering of our civilization.

REFERENCES

The subjects which we have attempted to discuss in this chapter are so complicated, and their ramifications are so many and varied, that the treatment we have given is of necessity quite superficial. We hope that we have explained at least in outline some of the fundamental limitations on the use of computing machines as they now exist or as they are likely to develop in the next few years. We have tried to make the point that although machines can do some things much better than men, there are many other things that men are always likely to do better than machines. The logical thing to do, therefore, is to try to exploit both machines and men to the best advantage. Since this book was written, Dr. W. Ross Ashby's important book *Design for a Brain* has appeared, and the reader who is interested in the relation between brain and machine should read this and Professor J. Z. Young's book *Doubt and Certainty in Science.*

1. *Proceedings of the Second International Congress on Orthopedagogics* (July, 1949). Systemen Keesing (Amsterdam, 1951).
2. Hartree, D. R. *Calculating Instruments and Machines.* Cambridge (1950).
3. Oettinger, A. G. "Programming a Digital Computer to Learn." *Phil. Mag.* (In the Press.)
4. Turing, A. M. *Mind: A quarterly review of Psychology and Philosophy.* 59 N.S., 236 (Oct., 1950), 433.
5. Watson-Watt, Sir R. *The Hibbert Journal,* **48** (Oct., 1949) 8.
6. *Cerebral Mechanisms and Behaviour,* The Hickson Symposium. Chapman and Hall (1950).
7. Bush, V. *The Atlantic Monthly* (July, 1945).
8. Herbert, Sir A. P. *Misleading Cases in the Common Law.*

APPENDIXES

APPENDIX 1

The following text is reprinted directly from pages 666–731 of *Taylor's Scientific Memoirs*, Vol. III. The editorial notes are by the translator, the Countess of Lovelace.

Article XXIX

Sketch of the Analytical Engine invented by Charles Babbage, *Esq. By* L. F. Menabrea, *of Turin, Officer of the Military Engineers.*

[From the *Bibliothèque Universelle de Genève,* No. 82. October, 1842]

[Before submitting to our readers the translation of M. Menabrea's memoir "On the Mathematical Principles of the Analytical Engine" invented by Mr. Babbage, we shall present to them a list of the printed papers connected with the subject, and also of those relating to the Difference Engine by which it was preceded.

For information on Mr. Babbage's "*Difference* Engine," which is but slightly alluded to by M. Menabrea, we refer the reader to the following sources—

1. Letter to Sir Humphry Davy, Bart., P.R.S., on the Application of Machinery to Calculate and Print Mathematical Tables. By Charles Babbage, Esq., F.R.S. London, July 1822. Reprinted, with a Report of the Council of the Royal Society, by order of the House of Commons, May, 1823.

2. On the Application of Machinery to the Calculation of Astronomical and Mathematical Tables. By Charles Babbage, Esq.—Memoirs of the Astronomical Society, vol. i, part 2. London, 1822.

3. Address to the Astronomical Society by Henry Thomas Colebrooke, Esq., F.R.S., President, on presenting the first Gold Medal of the Society to Charles Babbage, Esq., for the invention of the Calculating Engine.—Memoirs of the Astronomical Society. London, 1822.

4. On the Determination of the General Term of a New Class of Infinite Series. By Charles Babbage, Esq.—Transactions of the Cambridge Philosophical Society.

5. On Mr. Babbage's New Machine for Calculating and Printing Mathematical Tables.—Letter from Francis Baily, Esq., F.R.S., to M. Schumacher. No. 46, Astronomische Nachrichten. Reprinted in the Philosophical Magazine, May, 1824.

6. On a Method of expressing by Signs the Action of Machinery. By Charles Babbage, Esq.—Philosophical Transactions. London, 1826.

7. On Errors common to many Tables of Logarithms. By Charles Babbage, Esq.—Memoirs of the Astronomical Society, London, 1827.

8. Report of the Committee appointed by the Council of the Royal Society to consider the subject referred to in a communication received by

them from the Treasury respecting Mr. Babbage's Calculating Engine, and to report thereon. London, 1829.

9. Economy of Manufactures, chap. xx. 8vo. London, 1832.

10. Article on Babbage's Calculating Engine—Edinburgh Review, July, 1834. No. 120, vol. lix.

The present state of the Difference Engine, which has always been the property of Government, is as follows: The drawings are nearly finished, and the mechanical notation of the whole, recording every motion of which it is susceptible, is completed. A part of that Engine, comprising sixteen figures, arranged in three orders of differences, has been put together, and has frequently been used during the last eight years. It performs its work with absolute precision. This portion of the Difference Engine, together with all the drawings, are at present deposited in the Museum of King's College, London.

Of the Analytical Engine, which forms the principal object of the present memoir, we are not aware that any notice has hitherto appeared, except a Letter from the Inventor to M. Quetelet, Secretary to the Royal Academy of Sciences at Brussels, by whom it was communicated to that body. We subjoin a translation of this Letter, which was itself a translation of the original, and was not intended for publication by its author.

Royal Academy of Sciences at Brussels. General Meeting of the 7th and 8th of May, 1835

"A Letter from Mr. Babbage announces that he has for six months been engaged in making the drawings of a new calculating machine of far greater power than the first.

" 'I am myself astonished,' says Mr. Babbage, 'at the power I have been enabled to give to this machine; a year ago I should not have believed this result possible. This machine is intended to contain a hundred variables (or numbers susceptible of changing); each of these numbers may consist of twenty-five figures, $v_1, v_2, \ldots v_n$ being any numbers whatever, n being less than a hundred; if $f(v_1, v_2, v_3, \ldots v_n)$ be any given function which can be formed by addition, subtraction, multiplication, division, extraction of roots, or elevation to powers, the machine will calculate its numerical value; it will afterwards substitute this value in the place of v, or of any other variable, and will calculate this second function with respect to v. It will reduce to tables almost all equations of finite differences. Let us suppose that we have observed a thousand values of a, b, c, d, and that we wish to calculate them by the formula

$$p = \sqrt{\frac{a+b}{cd}}$$

the machine must be set to calculate the formula; the first series of the values of a, b, c, d must be adjusted to it; it will then calculate them, print them, and reduce them to zero; lastly, it will ring a bell to give notice that a new set of constants must be inserted. When there exists a relation between any number of successive coefficients of a series, provided it can be expressed as has already been said, the machine will calculate them and make their terms known in succession; it may afterwards be disposed so as to find the value of the series for all the values of the variable.'

"Mr. Babbage announces, in conclusion, 'that the greatest difficulties of the invention have already been surmounted, and that the plans will be finished in a few months.' "

In the Ninth Bridgewater Treatise, Mr. Babbage has employed several arguments deduced from the Analytical Engine, which afford some idea of its powers. See Ninth Bridgewater Treatise, 8vo, second edition. London, 1834.

Some of the numerous drawings of the Analytical Engine have been engraved on wooden blocks, and from these (by a mode contrived by Mr. Babbage) various stereotype plates have been taken. They comprise—

1. Plan of the figure wheels for one method of adding numbers.

2. Elevation of the wheels and axis of ditto.

3. Elevation of framing only of ditto.

4. Section of adding wheels and framing together.

5. Section of the adding wheels, sign wheels and framing complete.

6. Impression from the original wood block.

7. Impressions from a stereotype cast of No. 6, with the letters and signs inserted. Nos. 2, 3, 4 and 5 were stereotypes taken from this.

8. Plan of adding wheels and of long and short pinions, by means of which *stepping* is accomplished.

N.B. This process performs the operation of multiplying or dividing a number by any power of ten.

9. Elevation of long pinions in the position for addition.

10. Elevation of long pinions in the position for stepping.

11. Plan of mechanism for carrying the tens (by anticipation), connected with long pinions.

12. Section of the chain of wires for anticipating carriage.

13. Sections of the elevation of parts of the preceding carriage.

All these were executed about five years ago. At a later period (August, 1840) Mr. Babbage caused one of his general plans (No. 25) of the whole Analytical Engine to be lithographed at Paris.

Although these illustrations have not been published, on account of the time which would be required to describe them, and the rapid succession of improvements made subsequently, yet copies have been freely given to many of Mr. Babbage's friends, and were in August, 1838, presented at Newcastle to the British Association for the Advancement of Science, and in August, 1840, to the Institute of France through M. Arago, as well as the Royal Academy of Turin through M. Plana—EDITOR.]

THOSE labours which belong to the various branches of the mathematical sciences, although on first consideration they seem to be the exclusive province of intellect, may, nevertheless, be divided into two distinct sections; one of which may be called the mechanical, because it is subjected to precise and invariable laws, that are capable of being expressed by means of the operations of matter; while the other, demanding the intervention of reasoning, belongs more specially to the domain of the understanding. This admitted, we may propose to execute, by means of machinery, the mechanical branch of these labours, reserving for pure intellect that which depends on the reasoning faculties. Thus the rigid exactness of those laws which regulate numerical calculations must frequently have suggested the employment of material instruments, either for executing the whole of such calculations or for abridging them; and thence have arisen several inventions having this object in view, but which have in general but partially attained it. For instance, the much-admired machine of Pascal is now simply an object of curiosity, which, whilst it displays the powerful intellect of its inventor, is yet of little utility in itself. Its powers extended no further than the execution of the four* first operations of arithmetic, and indeed were in reality confined to that of the two first, since multiplication and division were the result of a series of additions and subtractions. The chief drawback hitherto on most of such machines is, that they require the continual intervention of a human agent to regulate their movements, and thence arises a source of errors; so that, if their use has not become general for large numerical calculations, it is because they have not in fact resolved the double problem which the question presents, that of *correctness* in the results, united with *œconomy* of time.

Struck with similar reflections, Mr. Babbage has devoted some years to the realization of a gigantic idea. He proposed to himself nothing less

* This remark seems to require further comment, since it is in some degree calculated to strike the mind as being at variance with the subsequent passage (page 349), where it is explained that *an engine which can effect these four operations* can in fact effect *every species of calculation*. The apparent discrepancy is stronger too in the translation than in the original, owing to its being impossible to render precisely into the English tongue all the niceties of distinction which the French idiom happens to admit of in the phrases used for the two passages we refer to. The explanation lies in this: that in the one case the execution of these four operations is the *fundamental starting-point*, and the object proposed for attainment by the machine is the *subsequent combination of these* in every possible variety; whereas in the other case the execution of some *one* of these four operations, selected at pleasure, is the *ultimatum*, the sole and utmost result that can be proposed for attainment by the machine referred to, and which result it cannot any further combine or work upon. The one *begins* where the other *ends*. Should this distinction not now appear perfectly clear, it will become so on perusing the rest of the Memoir, and the Notes that are appended to it. —NOTE BY TRANSLATOR.

than the construction of a machine capable of executing not merely arithmetical calculations, but even all those of analysis, if their laws are known. The imagination is at first astounded at the idea of such an undertaking; but the more calm reflection we bestow on it, the less impossible does success appear, and it is felt that it may depend on the discovery of some principle so general, that if applied to machinery, the latter may be capable of mechanically translating the operations which may be indicated to it by algebraical notation. The illustrious inventor having been kind enough to communicate to me some of his views on this subject during a visit he made at Turin, I have, with his approbation, thrown together the impressions they have left on my mind. But the reader must not expect to find a description of Mr. Babbage's engine; the comprehension of this would entail studies of much length; and I shall endeavour merely to give an insight into the end proposed, and to develop the principles on which its attainment depends.

I must first premise that this engine is entirely different from that of which there is a notice in the "Treatise on the Œconomy of Machinery," by the same author. But as the latter gave rise* to the idea of the engine in question, I consider it will be a useful preliminary briefly to recall what were Mr. Babbage's first essays, and also the circumstances in which they originated.

It is well known that the French government, wishing to promote the extension of the decimal system, had ordered the construction of logarithmical and trigonometrical tables of enormous extent. M. de Prony, who had been entrusted with the direction of this undertaking, divided it into three sections, to each of which were appointed a special class of persons. In the first section the formulae were so combined as to render them subservient to the purposes of numerical calculation; in the second, these same formulae were calculated for values of the variable, selected at certain successive distances; and under the third section, comprising about eighty individuals, who were most of them only acquainted with the two first rules of arithmetic, the values which were intermediate to those calculated by the second section were interpolated by means of simple additions and subtractions.

An undertaking similar to that just mentioned having been entered upon in England, Mr. Babbage conceived that the operations performed under the third section might be executed by a machine; and this idea he

* The idea that the one engine is the offspring and has grown out of the other, is an exceedingly natural and plausible supposition, until reflection reminds us that no *necessary* sequence and connexion need exist between two such inventions, and that they *may* be wholly independent. M. Menabrea has shared this idea in common with persons who have not his profound and accurate insight into the nature of either engine. In Note A (see the Notes at the end of the Memoir) it will be found sufficiently explained, however, that this supposition is unfounded. M. Menabrea's opportunities were by no means such as could be adequate to afford him information on a point like this, which would be naturally and almost unconsciously *assumed,* and would scarcely suggest any inquiry with reference to it.—NOTE BY TRANSLATOR.

realized by means of mechanism, which has been in part put together, and to which the name Difference Engine is applicable, on account of the principle upon which its construction is founded. To give some notion of this, it will suffice to consider the series of whole square numbers, 1, 4, 9, 16, 25, 36, 49, 64, etc. By subtracting each of these from the succeeding one, we obtain a new series, which we will name the Series of First Differences, consisting of the numbers 3, 5, 7, 9, 11, 13, 15, etc. On subtracting from each of these the preceding one, we obtain the Second Differences, which are all constant and equal to 2. We may represent this succession of operations, and their results, in the following table—

	A Column of Square Numbers	B First Differences	C Second Differences
	1		
		3	
	4		2*b*
		5	
a	9		2*d*
		7	
c	16		2
		9	
	25		2
		11	
	36		

From the mode in which the two last columns B and C have been formed, it is easy to see that if, for instance, we desire to pass from the number 5 to the succeeding one 7, we must add to the former the constant difference 2; similarly, if from the square number 9 we would pass to the following one 16, we must add to the former the difference 7, which difference is in other words the preceding difference 5, plus the constant difference 2; or again, which comes to the same thing, to obtain 16 we have only to add together the three numbers 2, 5, 9, placed obliquely in the direction *ba*. Similarly, we obtain the number 25 by summing up the three numbers placed in the oblique direction *dc*: commencing by the addition 2 + 7, we have the first difference 9 consecutively to 7; adding 16 to the 9 we have the square 25. We see then that the three numbers 2, 5, 9 being given, the whole series of successive square numbers, and that of their first differences likewise, may be obtained by means of simple additions.

Now, to conceive how these operations may be reproduced by a machine, suppose the latter to have three dials, designated as *A*, *B*, *C*, on each of which are traced, say a thousand divisions, by way of example, over which a needle shall pass. The two dials, *C*, *B*, shall have in addition a registering hammer, which is to give a number of strokes equal to that of the divisions indicated by the needle. For each stroke of the registering hammer of the dial *C*, the needle *B* shall advance one division; similarly, the needle *A* shall advance one division for every stroke of the registering hammer of the dial *B*. Such is the general disposition of the mechanism.

This being understood, let us at the beginning of the series of operations we wish to execute, place the needle *C* on the division 2, the needle *B* on the division 5, and the needle *A* on the division 9. Let us allow the hammer of the dial *C* to strike; it will strike twice, and at the same time the needle *B* will pass over two divisions. The latter will then indicate the number 7, which succeeds the number 5 in the column of first differences. If we now permit the hammer of the dial *B* to strike in its turn, it will strike seven times, during which the needle *A* will advance seven divisions; these added to the nine already marked by it, will give the number 16, which is the square number consecutive to 9. If we now recommence these operations, beginning with the needle *C*, which is always to be left on the division 2, we shall perceive that by repeating them indefinitely, we may successively reproduce the series of whole square numbers by means of a very simple mechanism.

The theorem on which is based the construction of the machine we have just been describing, is a particular case of the following more general theorem: that if in any polynomial whatever, the highest power of whose variable is m, this same variable be increased by equal degrees; the corresponding values of the polynomial then calculated, and the first, second, third, etc., differences of these be taken (as for the preceding series of squares); the mth differences will all be equal to each other. So that, in order to reproduce the series of values of the polynomial by means of a machine analogous to the one above described, it is sufficient that there be $(m + 1)$ dials, having the mutual relations we have indicated. As the differences may be either positive or negative, the machine will have a contrivance for either advancing or retrograding each needle, according as the number to be algebraically added may have the sign *plus* or *minus*.

If from a polynomial we pass to a series having an infinite number of terms, arranged according to the ascending powers of the variable, it would at first appear, that in order to apply the machine to the calculation of the function represented by such a series, the mechanism must include an infinite number of dials, which would in fact render the thing impossible. But in many cases the difficulty will disappear, if we observe that for a great number of functions the series which represent them may be rendered convergent; so that, according to the degree of approximation desired, we may limit ourselves to the calculation of a certain number of terms of the series, neglecting the rest. By this method the question is reduced to the primitive case of a finite polynomial. It is thus that we can calculate the succession of the logarithms of numbers. But since, in this particular instance, the terms which had been originally neglected receive increments in a ratio so continually increasing for equal increments of the variable, that the degree of approximation required would ultimately be affected, it is necessary, at certain intervals, to calculate the value of the function by different methods, and then respectively to use the results thus obtained, as data whence to deduce, by means of the machine, the other intermediate values. We see that the machine here performs the office of the third section of calculators mentioned in describing the tables computed by order of the French government, and that the end originally proposed is thus fulfilled by it.

Such is the nature of the first machine which Mr. Babbage conceived. We see that its use is confined to cases where the numbers required are such as can be obtained by means of simple additions or subtractions; that the machine is, so to speak, merely the expression of one* particular theorem of analysis; and that, in short, its operations cannot be extended so as to embrace the solution of an infinity of other questions included within the domain of mathematical analysis. It was while contemplating the vast field which yet remained to be traversed, that Mr. Babbage, renouncing his original essays, conceived the plan of another system of mechanism whose operations should themselves possess all the generality of algebraical notation, and which, on this account, he denominates the *Analytical Engine.*

Having now explained the state of the question, it is time for me to develop the principle on which is based the construction of this latter machine. When analysis is employed for the solution of any problem, there are usually two classes of operations to execute: first, the numerical calculation of the various coefficients; and secondly, their distribution in relation to the quantities affected by them. If, for example, we have to obtain the product of two binomials $(a + bx)$ $(m + nx)$, the result will be represented by $am + (an + bm)x + bnx^2$, in which expression we must first calculate am, an, bm, bn; then take the sum of $an + bm$; and lastly, respectively distribute the coefficients thus obtained amongst the powers of the variable. In order to reproduce these operations by means of a machine, the latter must therefore possess two distinct sets of powers: first, that of executing numerical calculations; secondly, that of rightly distributing the values so obtained.

But if human intervention were necessary for directing each of these partial operations, nothing would be gained under the heads of correctness and œconomy of time; the machine must therefore have the additional requisite of executing by itself all the successive operations required for the solution of a problem proposed to it, when once the *primitive numerical data*

* See Note A.

for this same problem have been introduced. Therefore, since from the moment that the nature of the calculation to be executed or of the problem to be resolved have been indicated to it, the machine is, by its own intrinsic power, of itself to go through all the intermediate operations which lead to the proposed result, it must exclude all methods of trial and guess-work, and can only admit the direct processes of calculation.*

It is necessarily thus; for the machine is not a thinking being, but simply an automaton which acts according to the laws imposed upon it. This being fundamental, one of the earliest researches its author had to undertake, was that of finding means for effecting the division of one number by another without using the method of guessing indicated by the usual rules of arithmetic. The difficulties of effecting this combination were far from being among the least; but upon it depended the success of every other. Under the impossibility of my here explaining the process through which this end is attained, we must limit ourselves to admitting that the four first operations of arithmetic, that is addition, subtraction, multiplication and division, can be performed in a direct manner through the intervention of the machine. This granted, the machine is thence capable of performing every species of numerical calculation, for all such calculations ultimately resolve themselves into the four operations we have just named. To conceive how the machine can now go through its functions according to the laws laid down, we will begin by giving an idea of the manner in which it materially represents numbers.

Let us conceive a pile or vertical column consisting of an indefinite number of circular discs, all pierced through their centres by a common axis, around which each of them can take an independent rotatory movement. If round the edge of each of these discs are written the ten figures which constitute our numerical alphabet, we may then, by arranging a series of these figures in the same vertical line, express in this manner any number whatever. It is sufficient for this purpose that the first disc represent units, the second tens, the third hundreds, and so on. When two numbers have been thus written on two distinct columns, we may propose to combine them arithmetically with each other, and to obtain the result on a third column. In general, if we have a series of columns† consisting of discs, which columns we will designate as V_0, V_1, V_2, V_3, V_4, etc., we may require, for instance, to divide the number written on the column V_1 by that on the column V_4, and to obtain the result on the column V_7. To effect this operation, we must impart to the machine two distinct arrangements; through the first it is prepared for executing *a division*, and through the second the columns it is to operate on are indicated to it, and also the

* This must not be understood in too unqualified a manner. The engine is capable, under certain circumstances, of feeling about to discover which of two or more possible contingencies has occurred, and of then shaping its future course accordingly.—Note by Translator.

† See Note B.

column on which the result is to be represented. If this division is to be followed, for example, by the addition of two numbers taken on other columns, the two original arrangements of the machine must be simultaneously altered. If, on the contrary, a series of operations of the same nature is to be gone through, then the first of the original arrangements will remain, and the second alone must be altered. Therefore, the arrangements that may be communicated to the various parts of the machine, may be distinguished into two principal classes:

First, that relative to the *Operations*.

Secondly, that relative to the *Variables*.

By this latter we mean that which indicates the columns to be operated on. As for the operations themselves, they are executed by a special apparatus, which is designated by the name of *mill*, and which itself contains a certain number of columns, similar to those of the Variables. When two numbers are to be combined together, the machine commences by effacing them from the columns where they are written, that is it places *zero** on every disc of the two vertical lines on which the numbers were represented; and it transfers the numbers to the mill. There, the apparatus having been disposed suitably for the required operation, this latter is effected, and, when completed, the result itself is transferred to the column of Variables which shall have been indicated. Thus the mill is that portion of the machine which works, and the columns of Variables constitute that where the results are represented and arranged. After the preceding explanations, we may perceive that all fractional and irrational results will be represented in decimal fractions. Supposing each column to have forty discs, this extension will be sufficient for all degrees of approximation generally required.

It will now be inquired how the machine can of itself, and without having recourse to the hand of man, assume the successive dispositions suited to the operations. The solution of this problem has been taken from Jacquard's apparatus,† used for the manufacture of brocaded stuffs, in the following manner—

Two species of threads are usually distinguished in woven stuffs; one is the *warp* or longitudinal thread, the other the *woof* or transverse thread, which is conveyed by the instrument called the shuttle, and which crosses the longitudinal thread or warp. When a brocaded stuff is required, it is necessary in turn to prevent certain threads from crossing the woof, and this according to a succession which is determined by the nature of the design that is to be reproduced. Formerly this process was lengthy and difficult, and it was requisite that the workman, by attending to the design which he was to copy, should himself regulate the movements the threads

* Zero is not *always* substituted when a number is transferred to the mill. This is explained further on in the memoir, and still more fully in Note D.—NOTE BY TRANSLATOR.

† See Note C.

were to take. Thence arose the high price of this description of stuffs, especially if threads of various colours entered into the fabric. To simplify this manufacture, Jacquard devised the plan of connecting each group of threads that were to act together, with a distinct lever belonging exclusively to that group. All these levers terminate in rods, which are united together in one bundle, having usually the form of a parallelepiped with a rectangular base. The rods are cylindrical, and are separated from each other by small intervals. The process of raising the threads is thus resolved into that of moving these various lever-arms in the requisite order. To effect this, a rectangular sheet of pasteboard is taken, somewhat larger in size than a section of the bundle of lever-arms. If this sheet be applied to the base of the bundle, and an advancing motion be then communicated to the pasteboard, this latter will move with it all the rods of the bundle, and consequently the threads that are connected with each of them. But if the pasteboard, instead of being plain, were pierced with holes corresponding to the extremities of the levers which meet it, then, since each of the levers would pass through the pasteboard during the motion of the latter, they would all remain in their places. We thus see that it is easy so to determine the position of the holes in the pasteboard, that, at any given moment, there shall be a certain number of levers, and consequently of parcels of threads, raised, while the rest remain where they were. Supposing this process is successively repeated according to a law indicated by the pattern to be executed, we perceive that this pattern may be reproduced on the stuff. For this purpose we need merely compose a series of cards according to the law required, and arrange them in suitable order one after the other; then, by causing them to pass over a polygonal beam which is so connected as to turn a new face for every stroke of the shuttle, which face shall then be impelled parallelly to itself against the bundle of lever-arms, the operation of raising the threads will be regularly performed. Thus we see that brocaded tissues may be manufactured with a precision and rapidity formerly difficult to obtain.

Arrangements analogous to those just described have been introduced into the Analytical Engine. It contains two principal species of cards: first, Operation cards, by means of which the parts of the machine are so disposed as to execute any determinate series of operations, such as additions, subtractions, multiplications, and divisions; secondly, cards of the Variables, which indicate to the machine the columns on which the results are to be represented. The cards, when put in motion, successively arrange the various portions of the machine according to the nature of the processes that are to be effected, and the machine at the same time executes these processes by means of the various pieces of mechanism of which it is constituted.

In order more perfectly to conceive the thing, let us select as an example the resolution of two equations of the first degree with two unknown quantities. Let the following be the two equations, in which x and y are the unknown quantities—

$$\begin{cases} mx + ny = d \\ m'x + n'y = d' \end{cases}$$

We deduce $x = \frac{dn' - d'n}{n'm - nm'}$, and for y an analogous expression. Let us continue to represent by V_0, V_1, V_2, etc. the different columns which contain the numbers, and let us suppose that the first eight columns have been chosen for expressing on them the numbers represented by m, n, d, m', n', d', n and n', which implies that $V_0 = m$, $V_1 = n$, $V_2 = d$, $V_3 = m'$, $V_4 = n'$, $V_5 = d'$, $V_6 = n$, $V_7 = n'$.

The series of operations commanded by the cards, and the results obtained, may be represented in the following table—

Number of the Operations	Operation-cards	Cards of the variables		Progress of the Operations
	Symbols Indicating the Nature of the Operations	Columns on which Operations are to be Performed	Columns which Receive Results of Operations	
1	×	$V_2 \times V_4 =$	V_8 . .	$= dn'$
2	×	$V_5 \times V_1 =$	V_9 . .	$= d'n$
3	×	$V_4 \times V_0 =$	V_{10} . .	$= n'm$
4	×	$V_1 \times V_3 =$	V_{11} . .	$= nm'$
5	—	$V_8 - V_9 =$	V_{12} . .	$= dn' - d'n$
6	—	$V_{10} - V_{11} =$	V_{13} . .	$= n'm - nm'$
7	÷	$\frac{V_{12}}{V_{13}} =$	V_{14} . .	$= x = \frac{dn' - d'n}{n'm - nm'}$

Since the cards do nothing but indicate in what manner and on what columns the machine shall act, it is clear that we must still, in every particular case, introduce the numerical data for the calculation. Thus, in the example we have selected, we must previously inscribe the numerical values of m, n, d, m', n', d', in the order and on the columns indicated, after which the machine when put in action will give the value of the unknown quantity x for this particular case. To obtain the value of y, another series of operations analogous to the preceding must be performed. But we see that they will be only four in number, since the denominator of the expression for y, excepting the sign, is the same as that for x, and equal to $n'm - nm'$. In the preceding table it will be remarked that the column for operations indicates four successive *multiplications*, two *subtractions*, and one *division*. Therefore, if desired, we need only use three operation cards; to manage which, it is sufficient to introduce into the machine an apparatus which shall, after the first multiplication, for instance, retain the card which relates to this operation, and not allow it to advance so as to be replaced by another one, until after this same operation shall have been four times repeated. In the preceding example we have seen, that to find the value of x we must begin by writing the coefficients m, n, d, m', n', d', upon eight columns, thus repeating n and n' twice. According to the same method, if it were required to calculate y likewise, these coefficients must be written on twelve different columns. But it is possible to simplify this process, and thus to diminish the chances of errors, which chances are greater, the larger the number of the quantities that have to be inscribed previous to setting the machine in action. To understand this simplification, we must remember that every number written on a column must, in order to be arithmetically combined with another number, be effaced from the column on which it is, and transferred to the *mill*. Thus, in the example we have discussed, we will take the two coefficients m and n', which are each of them to enter into *two* different products, that is m into mn' and md', n' into mn' and $n'd$. These coefficients will be inscribed on the columns V_0 and V_4. If we commence the series of operations by the product of m into n', these numbers will be effaced from the columns V_0 and V_4, that they may be transferred to the mill, which will multiply them into each other, and will then command the machine to represent the result, say on the column V_6. But as these numbers are each to be used again in another operation, they must again be inscribed somewhere; therefore, while the mill is working out their product, the machine will inscribe them anew on any two columns that may be indicated to it through the cards; and, as in the actual case, there is no reason why they should not resume their former places, we will suppose them again inscribed on V_0 and V_4, whence in short they would not finally disappear, to be reproduced no more, until they should have gone through all the combinations in which they might have to be used.

We see, then, that the whole assemblage of operations requisite for resolving the two* above equations of the first degree, may be definitively represented in the table on page 354.

In order to diminish to the utmost the chances of error in inscribing the numerical data of the problem, they are successively placed on one of the columns of the mill; then, by means of cards arranged for this purpose, these same numbers are caused to arrange themselves on the requisite columns, without the operator having to give his attention to it; so that his undivided mind may be applied to the simple inscription of these same numbers.

According to what has now been explained, we see that the collection of columns of Variables may be regarded as a *store* of numbers, accumulated there by the mill, and which, obeying the orders transmitted to the

* See Note D.

machine by means of the cards, pass alternately from the mill to the store, and from the store to the mill, that they may undergo the transformations demanded by the nature of the calculation to be performed.

Columns on which are Inscribed the Primitive Data	Number of the Operations	Cards of the Operations		Variable Cards			Statement of Results
		Number of the Operation cards	Nature of each Operation	Columns Acted on by each Operation	Columns that Receive the Result of each Operation	Indication of Change of Value on any Column	
$^1V_0 = m$	1	1	×	$^1V_0 \times {}^1V_4 =$	$^1V_6 \ldots$	$\begin{Bmatrix} ^1V_0 = {}^1V_0 \\ ^1V_4 = {}^1V_4 \end{Bmatrix}$	$^1V_6 = mn'$
$^1V_1 = n$	2	,,	×	$^1V_3 \times {}^1V_1 =$	$^1V_7 \ldots$	$\begin{Bmatrix} ^1V_3 = {}^1V_3 \\ ^1V_1 = {}^1V_1 \end{Bmatrix}$	$^1V_7 = m'n$
$^1V_2 = d$	3	,,	×	$^1V_2 \times {}^1V_4 =$	$^1V_8 \ldots$	$\begin{Bmatrix} ^1V_2 = {}^1V_2 \\ ^1V_4 = {}^0V_4 \end{Bmatrix}$	$^1V_8 = dn'$
$^1V_3 = m'$	4	,,	×	$^1V_5 \times {}^1V_1 =$	$^1V_9 \ldots$	$\begin{Bmatrix} ^1V_5 = {}^1V_5 \\ ^1V_1 = {}^0V_1 \end{Bmatrix}$	$^1V_9 = d'n$
$^1V_4 = n'$	5	,,	×	$^1V_0 \times {}^1V_5 =$	$^1V_{10} \ldots$	$\begin{Bmatrix} ^1V_0 = {}^0V_0 \\ ^1V_5 = {}^0V_5 \end{Bmatrix}$	$^1V_{10} = d'm$
$^1V_5 = d'$	6	,,	×	$^1V_2 \times {}^1V_3 =$	$^1V_{11} \ldots$	$\begin{Bmatrix} ^1V_2 = {}^0V_2 \\ ^1V_3 = {}^0V_3 \end{Bmatrix}$	$^1V_{11} = dm'$
	7	2	—	$^1V_6 - {}^1V_7 =$	$^1V_{12} \ldots$	$\begin{Bmatrix} ^1V_6 = {}^0V_6 \\ ^1V_7 = {}^0V_7 \end{Bmatrix}$	$^1V_{12} = mn' - m'n$
	8	,,	—	$^1V_8 - {}^1V_9 =$	$^1V_{13} \ldots$	$\begin{Bmatrix} ^1V_8 = {}^0V_8 \\ ^1V_9 = {}^0V_9 \end{Bmatrix}$	$^1V_{13} = dn' - d'n$
	9	,,	—	$^1V_{10} - {}^1V_{11} =$	$^1V_{14} \ldots$	$\begin{Bmatrix} ^1V_{10} = {}^0V_{10} \\ ^1V_{11} = {}^0V_{11} \end{Bmatrix}$	$^1V_{14} = d'm - dm'$
	10	3	÷	$^1V_{13} \div {}^1V_{12} =$	$^1V_{15} \ldots$	$\begin{Bmatrix} ^1V_{13} = {}^0V_{13} \\ ^1V_{12} = {}^1V_{12} \end{Bmatrix}$	$^1V_{15} = \dfrac{dn' - d'n}{mn' - m'n} = x$
	11	,,	÷	$^1V_{14} \div {}^1V_{12} =$	$^1V_{16} \ldots$	$\begin{Bmatrix} ^1V_{14} = {}^0V_{14} \\ ^1V_{12} = {}^0V_{12} \end{Bmatrix}$	$^1V_{16} = \dfrac{d'm - dm'}{mn' - m'n} = y$
1	2	3	4	5	6	7	8

Hitherto no mention has been made of the *signs* in the results, and the machine would be far from perfect were it incapable of expressing and combining amongst each other positive and negative quantities. To accomplish this end, there is, above every column, both of the mill and of the store, a disc, similar to the discs of which the columns themselves consist. According as the digit on this disc is even or uneven, the number inscribed on the corresponding column below it will be considered as positive or negative. This granted, we may, in the following manner, conceive how the signs can be algebraically combined in the machine. When a number is to be transferred from the store to the mill, and vice versa, it will always be transferred with its sign, which will be effected by means of the cards, as has been explained in what precedes. Let any two numbers then, on which we are to operate arithmetically, be placed in the mill with their respective signs. Suppose that we are first to add them together; the operation-cards will command the addition: if the two numbers be of the same sign, one of the two will be entirely effaced from where it was inscribed, and will go to add itself on the column which contains the other number; the machine will, during this operation, be able, by means of a certain apparatus, to prevent any movement in the disc of signs which belongs to the column on which the addition is made, and thus the result will remain with the sign which the two given numbers originally had. When two numbers have two different signs, the addition commanded by the card will be changed into a subtraction through the intervention of mechanisms which are brought into play by this very difference of sign. Since the subtraction can only be effected on the larger of the two numbers, it must be arranged that the disc of signs of the larger number shall not move while the smaller of the two numbers is being effaced from its column and subtracted from the other, whence the result will have the sign of this latter, just as in fact it ought to be. The combinations to which algebraical subtraction give rise, are analogous to the preceding. Let us pass on to multiplication. When two numbers to be multiplied are of the same sign, the result is positive; if the signs are different, the product must be negative. In order that the machine may act conformably to this law, we have but to conceive that on the column containing the product of the two given numbers, the digit which indicates the sign of that product, has been formed by the mutual addition of the two digits that respectively indicated the signs of the two given numbers; it is then obvious that if the digits of the signs are both even, or both odd, their sum will be an even number, and consequently will express a positive number; but that if, on the contrary, the two digits of the signs are one even and the other odd, their sum will be an odd number, and will consequently express a negative number. In the case of division, instead of adding the digits of the discs, they must be subtracted one from the other, which will produce results analogous to the preceding; that is to say, that if these figures are both even or both uneven, the remainder of this subtraction will be even; and it will be uneven in the contrary case. When I speak of mutually adding or subtracting the numbers expressed by the digits of the signs, I merely mean that one of the sign-discs is made to advance or retrograde a number of divisions equal to that which is expressed by the digit on the other sign-disc. We see, then, from the preceding explanation, that it is possible mechanically to combine the signs of quantities so as to obtain results conformable to those indicated by algebra.*

The machine is not only capable of executing those numerical calculations which depend on a given algebraical formula, but it is also fitted for analytical calculations in which there are one or several variables to be considered. It must be assumed that the analytical expression to be operated on can be developed according to powers of the variable, or according to determinate functions of this same variable, such as circular functions, for instance; and similarly for the result that is to be attained. If we then suppose that above the columns of the store, we have inscribed the powers or the functions of the variable, arranged according to whatever is the prescribed law of development, the coefficients of these several terms may be respectively placed on the corresponding column below each. In this manner we shall have a representation of an analytical development; and, supposing the position of the several terms composing it to be invariable, the problem will be reduced to that of calculating their coefficients according to the laws demanded by the nature of the question. In order to make this more clear, we shall take the following† very simple example, in which we are to multiply $(a + bx^1)$ by $(A + B \cos^1 x)$. We shall begin by writing x^0, x^1, $\cos^0 x$, $\cos^1 x$, above the columns V_0, V_1, V_2, V_3; then, since from the form of the two functions to be combined, the terms which are to compose the products will be of the following nature, $x^0 . \cos^0 x$, $x^0 . \cos^1 x$, $x^1 . \cos^0 x$, $x^1 . \cos^1 x$; these will be inscribed above the columns V_4, V_5, V_6, V_7. The coefficients of x^0, x^1, $\cos^0 x$, $\cos^1 x$ being given, they will, by means of the mill, be passed to the columns V_0, V_1, V_2 and V_3. Such are the primitive data of the problem. It is now the business of the machine to work out its solution, that is to find the coefficients which are to be inscribed on V_4, V_5, V_6, V_7. To attain this object, the law of formation of these same coefficients being known, the machine will act through the intervention of the cards, in the manner indicated by the table on page 357.

It will now be perceived that a general application may be made of the principle developed in the preceding example, to every species of process which it may be proposed to effect on series submitted to calculation. It is sufficient that the law of formation of the coefficients be known, and that this law be inscribed on the cards of the machine, which will then of itself execute all the calculations requisite for arriving at the proposed result. If, for instance, a recurring series were proposed, the law of formation of the coefficients being here uniform, the same operations which must be

* Not having had leisure to discuss with Mr. Babbage the manner of introducing into his machine the combination of algebraical signs, I do not pretend here to expose the method he uses for this purpose; but I considered that I ought myself to supply the deficiency, conceiving that this paper would have been imperfect if I had omitted to point out one means that might be employed for resolving this essential part of the problem in question.

† See Note E.

performed for one of them will be repeated for all the others; there will merely be a change in the locality of the operation, that is it will be performed with different columns. Generally, since every analytical expression is susceptible of being expressed in a series ordered according to certain functions of the variable, we perceive that the machine will include all analytical calculations which can be definitively reduced to the formation of coefficients according to certain laws, and to the distribution of these with respect to the variables.

We may deduce the following important consequence from these explanations, viz. that since the cards only indicate the nature of the

* Columns above which are written the Functions of the Variable		Coefficients		Cards of the Operations		Cards of the Variables			
		Given	To be Formed	Number of the Operations	Nature of the Operation	Columns on which Operations are to be Performed	Columns on which are to be Inscribed the results of the Operations	Indication of change of Value on any Column submitted to an Operation	Results of the Operations
x^0	${}^1V_0$	a							
x^1	${}^1V_1$	b							
$\text{Cos}^0 x$	${}^1V_2$	A							
$\text{Cos}^1 x$	${}^1V_3$	B							
$x^0 \cos^0 x$	${}^0V_4$	—	aA	1	$\times$	${}^1V_0 \times {}^1V_2 =$	${}^1V_4$ . .	$\begin{cases} {}^1V_0 = {}^1V_0 \\ {}^1V_2 = {}^1V_2 \end{cases}$	${}^1V_4 = aA$ coefficients of $x^0 \cos^0 x$
$x^0 \cos^1 x$	${}^0V_5$	—	aB	2	$\times$	${}^1V_0 \times {}^1V_3 =$	${}^1V_5$ . .	$\begin{cases} {}^1V_0 = {}^0V_0 \\ {}^1V_3 = {}^1V_3 \end{cases}$	${}^1V_5 = aB$,, ,, $x^0 \cos^1 x$
$x^1 \cos^0 x$	${}^0V_6$	—	bA	3	$\times$	${}^1V_1 \times {}^1V_2 =$	${}^1V_6$ . .	$\begin{cases} {}^1V_1 = {}^1V_1 \\ {}^1V_2 = {}^0V_2 \end{cases}$	${}^1V_6 = bA$,, ,, $x^1 \cos^0 x$
$x^1 \cos^1 x$	${}^0V_7$	—	bB	4	$\times$	${}^1V_1 \times {}^1V_3 =$	${}^1V_7$ . .	$\begin{cases} {}^1V_1 = {}^0V_1 \\ {}^1V_3 = {}^0V_3 \end{cases}$	${}^1V_7 = bB$,, ,, $x^1 \cos^1 x$

operations to be performed, and the columns of Variables with which they are to be executed, these cards will themselves possess all the generality of analysis, of which they are in fact merely a translation. We shall now further examine some of the difficulties which the machine must surmount, if its assimilation to analysis is to be complete. There are certain functions which necessarily change in nature when they pass through zero or infinity, or whose values cannot be admitted when they pass these limits. When such cases present themselves, the machine is able, by means of a bell, to give notice that the passage through zero or infinity is taking place, and it then stops until the attendant has again set it in action for whatever process it may next be desired that it shall perform. If this process has been foreseen, then the machine, instead of ringing, will so dispose itself as to present the new cards which have relation to the operation that is to succeed the passage through zero and infinity. These new cards may

* For an explanation of the upper left-hand indices attached to the V's in this and in the preceding table, we must refer the reader to Note D, amongst those appended to the memoir.—Note by Translator.

follow the first, but may only come into play contingently upon one or other of the two circumstances just mentioned taking place.

Let us consider a term of the form ab^n; since the cards are but a translation of the analytical formula, their number in this particular case must be the same, whatever be the value of n; that is to say, whatever be the number of multiplications required for elevating b to the nth power (we are supposing for the moment that n is a whole number). Now, since the exponent n indicates that b is to be multiplied n times by itself, and all these operations are of the same nature, it will be sufficient to employ one single operation-card, viz. that which orders the multiplication.

But when n is given for the particular case to be calculated, it will be further requisite that the machine limit the number of its multiplications according to the given values. The process may be thus arranged. The three numbers a, b and n will be written on as many distinct columns of the store; we shall designate them V_0, V_1, V_2; the result ab^n will place itself on the column V_3. When the number n has been introduced into the machine, a card will order a certain registering-apparatus to mark $(n - 1)$, and will at the same time execute the multiplication of b by b. When this is completed, it will be found that the registering-apparatus has effaced a unit, and that it only marks $(n - 2)$; while the machine will now again order the number b written on the column V_1 to multiply itself with the product b^2 written on the column V_3, which will give b^3. Another unit is then effaced from the registering-apparatus, and the same processes are continually repeated until it only marks zero. Thus the number b^n will be found inscribed on V_3, when the machine, pursuing its course of operations, will order the product of b^n by a; and the required calculation will have been completed without there being any necessity that the number of operation-cards used should vary with the value of n. If n were negative, the cards, instead of ordering the multiplication of a by b^n, would order its division; this we can easily conceive, since every number, being inscribed with its respective sign, is consequently capable of reacting on the nature of the operations to be executed. Finally, if n were fractional, of the form $\frac{p}{q}$, an additional column would be used for the inscription of q, and the machine would bring into action two sets of processes, one for raising b to the power p, the other for extracting the qth root of the number so obtained.

Again, it may be required, for example, to multiply an expression of the form $ax^m + bx^n$ by another $Ax^p + Bx^q$, and then to reduce the product to the least number of terms, if any of the indices are equal. The two factors being ordered with respect to x, the general result of the multiplication would be $Aax^{m+p} + Abx^{n+p} + Bax^{m+q} + Bbx^{n+q}$. Up to this point the process presents no difficulties; but suppose that we have $m = p$ and $n = q$, and that we wish to reduce the two middle terms to a single one $(Ab + Ba)x^{m+q}$. For this purpose, the cards may order $m + q$ and $n + p$ to be transferred into the mill, and there subtracted one from the other; if the remainder is nothing, as would be the case on the present hypothesis, the mill will order other cards to bring to it the coefficients Ab and Ba, that it may add them together and give them in this state as a coefficient for the single term $x^{n+p} = x^{m+q}$.

This example illustrates how the cards are able to reproduce all the operations which intellect performs in order to attain a determinate result, if these operations are themselves capable of being precisely defined.

Let us now examine the following expression—

$$2 \cdot \frac{2^2 \cdot 4^2 \cdot 6^2 \cdot 8^2 \cdot 10^2 \ldots (2n)^2}{1^2 \cdot 3^2 \cdot 5^2 \cdot 7^2 \cdot 9^2 \ldots (2n-1)^2 \cdot (2n+1)^2}$$

which we know becomes equal to the ratio of the circumference to the diameter, when n is infinite. We may require the machine not only to perform the calculation of this fractional expression, but further to give indication as soon as the value becomes identical with that of the ratio of the circumference to the diameter when n is infinite, a case in which the computation would be impossible. Observe that we should thus require of the machine to interpret a result not of itself evident, and that this is not amongst its attributes, since it is no thinking being. Nevertheless, when the case of $n = \infty$ has been foreseen, a card may immediately order the substitution of the value of π (π being the ratio of the circumference to the diameter), without going through the series of calculations indicated. This would merely require that the machine contain a special card, whose office it should be to place the number π in a direct and independent manner on the column indicated to it. And here we should introduce the mention of a third species of cards, which may be called *cards of numbers*. There are certain numbers, such as those expressing the ratio of the circumference to the diameter, the Numbers of Bernoulli, etc., which frequently present themselves in calculations. To avoid the necessity for computing them every time they have to be used, certain cards may be combined specially in order to give these numbers ready made into the mill, whence they afterwards go and place themselves on those columns of the store that are destined for them. Through this means the machine will be susceptible of those simplifications afforded by the use of numerical tables. It would be equally possible to introduce, by means of these cards, the logarithms of numbers; but perhaps it might not be in this case either the shortest or the most appropriate method; for the machine might be able to perform the same calculations by other more expeditious combinations, founded on the rapidity with which it executes the four first operations of arithmetic. To give an idea of this rapidity, we need only mention that Mr. Babbage believes he can, by his engine, form the product of two numbers, each containing twenty figures, in *three minutes*.

Perhaps the immense number of cards required for the solution of any rather complicated problem may appear to be an obstacle; but this does not seem to be the case. There is no limit to the number of cards that can be used. Certain stuffs require for their fabrication not less than *twenty thousand* cards, and we may unquestionably far exceed even this quantity.*

Resuming what we have explained concerning the Analytical Engine, we may conclude that it is based on two principles: the first, consisting in the fact that every arithmetical calculation ultimately depends on four principal operations—addition, subtraction, multiplication, and division; the second, in the possibility of reducing every analytical calculation to that of the coefficients for the several terms of a series. If this last principle be true, all the operations of analysis come within the domain of the engine. To take another point of view: the use of the cards offers a generality equal to that of algebraical formulae, since such a formula simply indicates the nature and order of the operations requisite for arriving at a certain definite result, and similarly the cards merely command the engine to perform these same operations; but in order that the mechanisms may be able to act to any purpose, the numerical data of the problem must in every particular case be introduced. Thus the same series of cards will serve for all questions whose sameness of nature is such as to require nothing altered excepting the numerical data. In this light the cards are merely a translation of algebraical formulae, or, to express it better, another form of analytical notation.

Since the engine has a mode of acting peculiar to itself, it will in every particular case be necessary to arrange the series of calculations conformably to the means which the machine possesses; for such or such a process which might be very easy for a calculator, may be long and complicated for the engine, and vice versa.

Considered under the most general point of view, the essential object of the machine being to calculate, according to the laws dictated to it, the values of numerical coefficients which it is then to distribute appropriately on the columns which represent the variables, it follows that the interpretation of formulae and of results is beyond its province, unless indeed this very interpretation be itself susceptible of expression by means of the symbols which the machine employs. Thus, although it is not itself the being that reflects, it may yet be considered as the being which executes the conceptions of intelligence.† The cards receive the impress of these conceptions, and transmit to the various trains of mechanism composing the engine the orders necessary for their action. When once the engine shall have been constructed, the difficulty will be reduced to the making out of the cards; but as these are merely the translation of algebraical formulae, it will, by means of some simple notations, be easy to consign the execution of them to a workman. Thus the whole intellectual labour will

* See Note F. † See Note G.

be limited to the preparation of the formulae, which must be adapted for calculation by the engine.

Now, admitting that such an engine can be constructed, it may be inquired: what will be its utility? To recapitulate; it will afford the following advantages: first, rigid accuracy. We know that numerical calculations are generally the stumbling-block to the solution of problems, since errors easily creep into them, and it is by no means always easy to detect these errors. Now the engine, by the very nature of its mode of acting, which requires no human intervention during the course of its operations, presents every species of security under the head of correctness; besides, it carries with it its own check; for at the end of every operation it prints off, not only the results, but likewise the numerical data of the question; so that it is easy to verify whether the question has been correctly proposed. Secondly, economy of time: to convince ourselves of this, we need only recollect that the multiplication of two numbers, consisting each of twenty figures, requires at the very utmost three minutes. Likewise, when a long series of identical computations is to be performed, such as those required for the formation of numerical tables, the machine can be brought into play so as to give several results at the same time, which will greatly abridge the whole amount of the processes. Thirdly, economy of intelligence: a simple arithmetical computation requires to be performed by a person possessing some capacity; and when we pass to more complicated calculations, and wish to use algebraical formulae in particular cases, knowledge must be possessed which pre-supposes preliminary mathematical studies of some extent. Now the engine, from its capability of performing by itself all these purely material operations, spares intellectual labour, which may be more profitably employed. Thus the engine may be considered as a real manufactory of figures, which will lend its aid to those many useful sciences and arts that depend on numbers. Again, who can foresee the consequences of such an invention? In truth, how many precious observations remain practically barren for the progress of the sciences, because there are not powers sufficient for computing the results! And what discouragement does the perspective of a long and arid computation cast into the mind of a man of genius, who demands time exclusively for meditation, and who beholds it snatched from him by the material routine of operations! Yet it is by the laborious route of analysis that he must reach truth; but he cannot pursue this unless guided by numbers; for without numbers it is not given us to raise the veil which envelopes the mysteries of nature. Thus the idea of constructing an apparatus capable of aiding human weakness in such researches, is a conception which, being realized, would mark a glorious epoch in the history of the sciences. The plans have been arranged for all the various parts, and for all the wheel-work, which compose this immense apparatus, and their action studied; but these have not yet been fully combined together in the drawings* and mechanical notation.† The confidence which the genius of Mr. Babbage must inspire, affords legitimate ground for hope that this enterprise will be crowned with success; and while we render homage to the intelligence which directs it, let us breathe aspirations for the accomplishment of such an undertaking.

NOTES BY THE TRANSLATOR

Note A—Page 348

The particular function whose integral the Difference Engine was constructed to tabulate, is

$$\Delta^7 u_z = 0$$

The purpose which that engine has been specially intended and adapted to fulfil, is the computation of nautical and astronomical tables. The integral of

$$\Delta^7 u_z = 0$$

being $$u_z = a + bx + cx^2 + dx^3 + ex^4 + fx^5 + gx^6$$

the constants a, b, c, etc. are represented on the seven columns of discs, of which the engine consists. It can therefore tabulate *accurately* and to an *unlimited extent*, all series whose general term is comprised in the above formula; and it can also tabulate *approximately* between *intervals of greater or less extent*, all other series which are capable of tabulation by the Method of Differences.

The Analytical Engine, on the contrary, is not merely adapted for *tabulating* the results of one particular function and of no other, but for *developing and tabulating* any function whatever. In fact the engine may be described as being the material expression of any indefinite function of any degree of generality and complexity, such as for instance,

$$F(x, y, z, \log x, \sin y, x^p, \text{etc.})$$

which is, it will be observed, a function of all other possible functions of any number of quantities.

In this, which we may call the *neutral* or *zero* state of the engine, it is ready to receive at any moment, by means of cards constituting a portion

* This sentence has been slightly altered in the translation in order to express more exactly the present state of the engine.—Note by Translator.

† The notation here alluded to is a most interesting and important subject, and would have well deserved a separate and detailed note upon it, amongst those appended to the Memoir. It has, however, been impossible, within the space allotted, even to touch upon so wide a field.—Note by Translator.

of its mechanism (and applied on the principle of those used in the Jacquard loom), the impress of whatever *special* function we may desire to develop or to tabulate. These cards contain within themselves (in a manner explained in the Memoir itself, pages 349 and 350) the law of development of the particular function that may be under consideration, and they compel the mechanism to act accordingly in a certain corresponding order. One of the simplest cases would be, for example, to suppose that

$$F(x, y, z, \text{etc.})$$

is the particular function

$$\Delta^n u_z = 0$$

which the Difference Engine tabulates for values of n only up to 7. In this case the cards would order the mechanism to go through that succession of operations which would tabulate

$$u_z = a + bx + cx^2 + \ldots + mx^{n-1}$$

where n might be any number whatever.

These cards, however, have nothing to do with the regulation of the particular *numerical* data. They merely determine the *operations** to be effected, which operations may of course be performed on an infinite variety of particular numerical values, and do not bring out any definite numerical results unless the numerical data of the problem have been impressed on the requisite portions of the train of mechanism. In the above example, the first essential step towards an arithmetical result, would be the substitution of specific numbers for n, and for the other primitive quantities which enter into the function.

Again, let us suppose that for F we put two complete equations of the fourth degree between x and y. We must then express on the cards the law of elimination for such equations. The engine would follow out those laws, and would ultimately give the equation of one variable which results from such elimination. Various *modes* of elimination might be selected; and of course the cards must be made out accordingly. The following is one mode that might be adopted. The engine is able to multiply together any two functions of the form

$$a + bx + cx^2 + \ldots + px^n$$

This granted, the two equations may be arranged according to the powers of y, and the coefficients of the powers of y may be arranged according to

* We do not mean to imply that the *only* use made of the Jacquard cards is that of regulating the algebraical *operations*. But we mean to explain that *those* cards and portions of mechanism which regulate these *operations*, are wholly independent of those which are used for other purposes. M. Menabrea explains that there are *three* classes of cards used in the engine for three distinct sets of objects, viz. *Cards of the Operations*, *Cards of the Variables*, and certain *Cards of Numbers*. (See pages 351 and 353.)

powers of x. The elimination of y will result from the successive multiplications and subtractions of several such functions. In this, and in all other instances, as was explained above, the particular *numerical* data and the *numerical* results are determined by means and by portions of the mechanism which act quite independently of those that regulate the *operations.*

In studying the action of the Analytical Engine, we find that the peculiar and independent nature of the considerations which in all mathematical analysis belong to *operations,* as distinguished from *the objects operated upon* and from the *results* of the operations performed upon those objects, is very strikingly defined and separated.

It is well to draw attention to this point, not only because its full appreciation is essential to the attainment of any very just and adequate general comprehension of the powers and mode of action of the Analytical Engine, but also because it is one which is perhaps too little kept in view in the study of mathematical science in general. It is, however, impossible to confound it with other considerations, either when we trace the manner in which that engine attains its results, or when we prepare the data for its attainment of those results. It were much to be desired, that when mathematical processes pass through the human brain instead of through the medium of inanimate mechanism, it were equally a necessity of things that the reasonings connected with *operations* should hold the same just place as a clear and well-defined branch of the subject of analysis, a fundamental but yet independent ingredient in the science, which they must do in studying the engine. The confusion, the difficulties, the contradictions which, in consequence of a want of accurate distinctions in this particular, have up to even a recent period encumbered mathematics in all those branches involving the consideration of negative and impossible quantities, will at once occur to the reader who is at all versed in this science, and would alone suffice to justify dwelling somewhat on the point, in connexion with any subject so peculiarly fitted to give forcible illustration of it, as the Analytical Engine. It may be desirable to explain, that by the word *operation,* we mean *any process which alters the mutual relation of two or more things*, be this relation of what kind it may. This is the most general definition, and would include all subjects in the universe. In abstract mathematics, of course, operations alter those particular relations which are involved in the considerations of number and space, and the *results* of operations are those peculiar results which correspond to the nature of the subjects of operation. But the science of operations, as derived from mathematics more especially, is a science of itself, and has its own abstract truth and value; just as logic has its own peculiar truth and value, independently of the subjects to which we may apply its reasonings and processes. Those who are accustomed to some of the more modern views of the above subject, will know that a few fundamental relations being true, certain other combinations of relations must of necessity follow;

combinations unlimited in variety and extent if the deductions from the primary relations be carried on far enough. They will also be aware that one main reason why the separate nature of the science of operations has been little felt, and in general little dwelt on, is the *shifting* meaning of many of the symbols used in mathematical notation. First, the symbols of *operation* are frequently *also* the symbols of the *results* of operations. We may say that these symbols are apt to have both a *retrospective* and a *prospective* signification. They may signify either relations that are the consequence of a series of processes already performed, or relations that are yet to be effected through certain processes. Secondly, figures, the symbols of *numerical magnitude*, are frequently *also* the symbols of *operations*, as when they are the indices of powers. Wherever terms have a shifting meaning, independent sets of considerations are liable to become complicated together, and reasonings and results are frequently falsified. Now in the Analytical Engine the operations which come under the first of the above heads, are ordered and combined by means of a notation and of a train of mechanism which belong exclusively to themselves; and with respect to the second head, whenever numbers meaning *operations* and not *quantities* (such as the indices of powers), are inscribed on any column or set of columns, those columns immediately act in a wholly separate and independent manner, becoming connected with the *operating mechanism* exclusively, and re-acting upon this. They never come into combination with numbers upon any other columns meaning *quantities*; though, of course, if there are numbers meaning *operations* upon *n* columns, these may *combine amongst each other*, and will often be required to do so, just as numbers meaning *quantities* combine with each other in any variety It might have been arranged that all numbers meaning *operations* should have appeared on some separate portion of the engine from that which presents numerical *quantities*; but the present mode is in some cases more simple, and offers in reality quite as much distinctness when understood.

The operating mechanism can even be thrown into action independently of any object to operate upon (although of course no *result* could then be developed). Again, it might act upon other things besides *number*, were objects found whose mutual fundamental relations could be expressed by those of the abstract science of operations, and which should be also susceptible of adaptations to the action of the operating notation and mechanism of the engine. Supposing, for instance, that the fundamental relations of pitched sounds in the science of harmony and of musical composition were susceptible of such expression and adaptations, the engine might compose elaborate and scientific pieces of music of any degree of complexity or extent.

The Analytical Engine is an *embodying of the science of operations*, constructed with peculiar reference to abstract number as the subject of those operations. The Difference Engine is the embodying of *one particular and* *very limited set of operations*, which (see the notation used in note B) may be expressed thus (+, +, +, +, +, +), or thus, 6(+). Six repetitions of the one operation, +, is, in fact, the whole sum and object of that engine. It has seven columns, and a number on any column can add itself to a number on the next column to its *right-hand*. So that, beginning with the column farthest to the left, six additions can be effected, and the result appears on the seventh column, which is the last on the right-hand. The *operating* mechanism of this engine acts in as separate and independent a manner as that of the Analytical Engine; but being susceptible of only one unvarying and restricted combination, it has little force or interest in illustration of the distinct nature of the *science of operations*. The importance of regarding the Analytical Engine under this point of view will, we think, become more and more obvious, as the reader proceeds with M. Menabrea's clear and masterly article. The calculus of operations is likewise in itself a topic of so much interest, and has of late years been so much more written on and thought on than formerly, that any bearing which that engine, from its mode of constitution, may possess upon the illustration of this branch of mathematical science, should not be overlooked. Whether the inventor of this engine had any such views in his mind while working out the invention, or whether he may subsequently ever have regarded it under this phase, we do not know; but it is one that forcibly occurred to ourselves on becoming acquainted with the means through which analytical combinations are actually attained by the mechanism. We cannot forbear suggesting one practical result which it appears to us must be greatly facilitated by the independent manner in which the engine orders and combines its *operations*: we allude to the attainment of those combinations into which *imaginary quantities* enter. This is a branch of its processes into which we have not had the opportunity of inquiring, and our conjecture therefore as to the principle on which we conceive the accomplishment of such results may have been made to depend, is very probably not in accordance with the fact, and less subservient for the purpose than some other principles, or at least requiring the co-operation of others. It seems to us obvious, however, that where operations are so independent in their mode of acting, it must be easy by means of a few simple provisions and additions in arranging the mechanism, to bring out a *double* set of *results*, viz.: first, the *numerical magnitudes* which are the results of operations performed on *numerical data*. (These results are the *primary* object of the engine.) Secondly, the *symbolical results* to be attached to those numerical results, which symbolical results are not less the necessary and logical consequences of operations performed upon *symbolical data*, than are numerical results when the data are numerical.*

* In fact such an extension as we allude to, would merely constitute a further and more perfected development of any system introduced for making the proper combinations of the signs *plus* and *minus*. How ably M. Menabrea has touched on this restricted case is pointed out in Note B.

If we compare together the powers and the principles of construction of the Difference and of the Analytical Engines, we shall perceive that the capabilities of the latter are immeasurably more extensive than those of the former, and that they in fact hold to each other the same relationship as that of analysis to arithmetic. The Difference Engine can effect but one particular series of operations, viz. that required for tabulating the integral of the special function

$$\Delta^n u_z = 0$$

and as it can only do this for values of n up to 7,* it cannot be considered as being the most *general* expression even of *one particular* function, much less as being the expression of any and all possible functions of all degrees of generality. The Difference Engine can in reality (as has been already partly explained) do nothing but *add*; and any other processes, not excepting those of simple subtraction, multiplication and division, can be performed by it only just to that extent in which it is possible, by judicious mathematical arrangement and artifices, to reduce them to a *series of additions*. The method of differences is, in fact, a method of additions; and as it includes within its means a larger number of results attainable by *addition* simply, than any other mathematical principle, it was very appropriately selected as the basis on which to construct *an Adding Machine*, so as to give to the powers of such a machine the widest possible range. The Analytical Engine, on the contrary, can either add, subtract, multiply or divide with equal facility; and performs each of these four operations in a direct manner, without the aid of any of the other three. This one fact implies everything; and it is scarcely necessary to point out, for instance, that while the Difference Engine can merely *tabulate*, and is incapable of *developing*, the Analytical Engine can *either tabulate or develop*.

The former engine is in its nature strictly *arithmetical*, and the results it can arrive at lie within a very clearly defined and restricted range, while there is no finite line of demarcation which limits the powers of the Analytical Engine. These powers are co-extensive with our knowledge of the laws of analysis itself, and need be bounded only by our acquaintance with the latter. Indeed we may consider the engine as the *material and mechanical representative* of analysis, and that our actual working powers in this department of human study will be enabled more effectually than

* The machine might have been constructed so as to tabulate for a higher value of n than seven. Since, however, every unit added to the value of n increases the extent of the mechanism requisite, there would on this account be a limit beyond which it could not be practically carried. Seven is sufficiently high for the calculation of all ordinary tables.

The fact that, in the Analytical Engine, the same extent of mechanism suffices for the solution of $\Delta^n u_z = 0$, whether $n = 7$, $n = 100{,}000$, or $n =$ any number whatever, at once suggests how entirely distinct must be the *nature of the principles* through whose application matter has been enabled to become the working agent of abstract mental operations in each of these engines respectively; and it affords an equally obvious presumption, that in the case of the Analytical Engine, not only are those principles in themselves of a higher and more comprehensive description, but also such as must vastly extend the *practical* value of the engine whose basis they constitute.

heretofore to keep pace with our theoretical knowledge of its principles and laws, through the complete control which the engine gives us over the *executive manipulation* of algebraical and numerical symbols.

Those who view mathematical science not merely as a vast body of abstract and immutable truths, whose intrinsic beauty, symmetry and logical completeness, when regarded in their connexion together as a whole, entitle them to a prominent place in the interest of all profound and logical minds, but as possessing a yet deeper interest for the human race, when it is remembered that this science constitutes the language through which alone we can adequately express the great facts of the natural world, and those unceasing changes of mutual relationship which, visibly or invisibly, consciously or unconsciously to our immediate physical perceptions, are interminably going on in the agencies of the creation we live amidst: those who thus think on mathematical truth as the instrument through which the weak mind of man can most effectually read his Creator's works, will regard with especial interest all that can tend to facilitate the translation of its principles into explicit practical forms.

The distinctive characteristic of the Analytical Engine, and that which has rendered it possible to endow mechanism with such extensive faculties as bid fair to make this engine the executive right-hand of abstract algebra, is the introduction into it of the principle which Jacquard devised for regulating, by means of punched cards, the most complicated patterns in the fabrication of brocaded stuffs. It is in this that the distinction between the two engines lies. Nothing of the sort exists in the Difference Engine. We may say most aptly that the Analytical Engine *weaves algebraical patterns* just as the Jacquard-loom weaves flowers and leaves. Here, it seems to us, resides much more of originality than the Difference Engine can be fairly entitled to claim. We do not wish to deny to this latter all such claims. We believe that it is the only proposal or attempt ever made to construct a calculating machine *founded on the principle of successive orders of differences*, and capable of *printing off its own results*; and that this engine surpasses its predecessors, both in the extent of the calculations which it can perform, in the facility, certainty and accuracy with which it can effect them, and in the absence of all necessity for the intervention of human intelligence *during the performance of its calculations*. Its nature is, however, limited to the strictly arithmetical, and it is far from being the first or only scheme for constructing *arithmetical* calculating machines with more or less of success.

The bounds of *arithmetic* were, however, outstepped the moment the idea of applying the cards had occurred; and the Analytical Engine does not occupy common ground with mere "calculating machines." It holds a position wholly its own; and the considerations it suggests are most interesting in their nature. In enabling mechanism to combine together *general* symbols, in successions of unlimited variety and extent, a uniting link is established between the operations of matter and the abstract mental processes of the *most abstract* branch of mathematical science. A new, a vast, and a powerful language is developed for the future use of analysis, in which to wield its truths so that these may become of more speedy and accurate practical application for the purposes of mankind than the means hitherto in our possession have rendered possible. Thus not only the mental and the material, but the theoretical and the practical in the mathematical world, are brought into more intimate and effective connexion with each other. We are not aware of its being on record that anything partaking of the nature of what is so well designated the *Analytical* Engine has been hitherto proposed, or even thought of, as a practical possibility, any more than the idea of a thinking or of a reasoning machine.

We will touch on another point which constitutes an important distinction in the modes of operating of the Difference and Analytical Engines. In order to enable the former to do its business, it is necessary to put into its columns the series of numbers constituting the first terms of the several orders of differences for whatever is the particular table under consideration. The machine then works *upon* these as its data. But these data must themselves have been already computed through a series of calculations by a human head. Therefore that engine can only produce results depending on data which have been arrived at by the explicit and actual working out of processes that are in their nature different from any that come within the sphere of its own powers. In other words, an *analysing* process must have been gone through by a human mind in order to obtain the data upon which the engine then *synthetically* builds its results. The Difference Engine is in its character exclusively *synthetical*, while the Analytical Engine is equally capable of analysis or of synthesis.

It is true that the Difference Engine can calculate to a much greater extent with these few preliminary data, than the data themselves required for their own determination. The table of squares, for instance, can be calculated to any extent whatever, when the numbers *one* and *two* are furnished; and a very few differences computed at any part of a table of logarithms would enable the engine to calculate many hundreds or even thousands of logarithms. Still the circumstance of its requiring, as a previous condition, that any function whatever shall have been numerically worked out, makes it very inferior in its nature and advantages to an engine which, like the Analytical Engine, requires merely that we should know the *succession and distribution of the operations* to be performed; without there being any occasion,* in order to obtain data on which it can work, for our ever having gone through either the same particular operations which it is itself to effect, or any others. Numerical data must of course be given it, but they are mere arbitrary ones; not data that could only be

* This subject is further noticed in Note F.

arrived at through a systematic and necessary series of previous numerical calculations, which is quite a different thing.

To this it may be replied that an analysing process must equally have been performed in order to furnish the Analytical Engine with the necessary *operative* data; and that herein may also lie a possible source of error. Granted that the actual mechanism is unerring in its processes, the *cards* may give it wrong orders. This is unquestionably the case; but there is much less chance of error, and likewise far less expenditure of time and labour, where operations only, and the distribution of these operations, have to be made out, than where explicit numerical results are to be attained. In the case of the Analytical Engine we have undoubtedly to lay out a certain capital of analytical labour in one particular line; but this is in order that the engine may bring us in a much larger return in another line. It should be remembered also that the cards when once made out for any formula, have all the generality of algebra, and include an infinite number of particular cases.

We have dwelt considerably on the distinctive peculiarities of each of these engines, because we think it essential to place their respective attributes in strong relief before the apprehension of the public; and to define with clearness and accuracy the wholly different nature of the principles on which each is based, so as to make it self-evident to the reader (the mathematical reader at least) in what manner and degree the powers of the Analytical Engine transcend those of an engine, which, like the Difference Engine, can only work out such results as may be derived from *one restricted and particular series of processes*, such as those included in $\Delta^n u_z = 0$. We think this of importance, because we know that there exists considerable vagueness and inaccuracy in the mind of persons in general on the subject. There is a misty notion amongst most of those who have attended at all to it, that *two* "calculating machines" have been successively invented by the same person within the last few years; while others again have never heard but of the one original "calculating machine," and are not aware of there being any extension upon this. For either of these two classes of persons the above considerations are appropriate. While the latter require a knowledge of the fact that there *are two* such inventions, the former are not less in want of accurate and well-defined information on the subject. No very clear or correct ideas prevail as to the characteristics of each engine, or their respective advantages or disadvantages; and, in meeting with those incidental allusions, of a more or less direct kind, which occur in so many publications of the day, to these machines, it must frequently be matter of doubt *which* "calculating machine" is referred to, or whether *both* are included in the general allusion.

We are desirous likewise of removing two misapprehensions which we know obtain, to some extent, respecting these engines. In the first place it is very generally supposed that the Difference Engine, after it had been completed up to a certain point, *suggested* the idea of the Analytical Engine; and that the second is in fact the improved offspring of the first, and *grew out* of the existence of its predecessor, through some natural or else accidental combination of ideas suggested by this one. Such a supposition is in this instance contrary to the facts; although it seems to be almost an obvious inference, wherever two inventions, similar in their nature and objects, succeed each other closely in order of *time*, and strikingly in order of *value*; more especially when the same individual is the author of both. Nevertheless the ideas which led to the Analytical Engine occurred in a manner wholly independent of any that were connected with the Difference Engine. These ideas are indeed in their own intrinsic nature independent of the latter engine, and might equally have occurred had it never existed nor been even thought of at all.

The second of the misapprehensions above alluded to, relates to the well-known suspension, during some years past, of all progress in the construction of the Difference Engine. Respecting the circumstances which have interfered with the actual completion of either invention, we offer no opinion; and in fact are not possessed of the data for doing so, had we the inclination. But we know that some persons suppose these obstacles (be they what they may) to have arisen *in consequence* of the subsequent invention of the Analytical Engine while the former was in progress. We have ourselves heard it even *lamented* that an idea should ever have occurred at all, which had turned out to be merely the means of arresting what was already in a course of successful execution, without substituting the superior invention in its stead. This notion we can contradict in the most unqualified manner. The progress of the Difference Engine had long been suspended, before there were even the least crude glimmerings of any invention superior to it. Such glimmerings, therefore, and their subsequent development, were in no way the original *cause* of that suspension; although, where difficulties of some kind or other evidently already existed, it was not perhaps calculated to remove or lessen them that an invention should have been meanwhile thought of, which, while including all that the first was capable of, possesses powers so extended as to eclipse it altogether.

We leave it for the decision of each individual (*after he has possessed himself* of competent information as to the characteristics of each engine), to determine how far it ought to be matter of regret that such an accession has been made to the powers of human science, even if it *has* (which we greatly doubt) increased to a certain limited extent some already existing difficulties that had arisen in the way of completing a valuable but lesser work. We leave it for each to satisfy himself as to the wisdom of desiring the obliteration (were that now possible) of all records of the more perfect invention, in order that the comparatively limited one might be finished.

The Difference Engine would doubtless fulfil all those practical objects which it was originally destined for. It would certainly calculate all the tables that are more directly necessary for the physical purposes of life, such as nautical and other computations. Those who incline to very strictly utilitarian views, may perhaps feel that the peculiar powers of the Analytical Engine bear upon questions of abstract and speculative science, rather than upon those involving everyday and ordinary human interests. These persons being likely to possess but little sympathy, or possibly acquaintance, with any branches of science which they do not find to be *useful* (according to *their* definition of that word), may conceive that the undertaking of that engine, now that the other one is already in progress, would be a barren and unproductive laying out of yet more money and labour; in fact, a work of supererogation. Even in the utilitarian aspect, however, we do not doubt that very valuable practical results would be developed by the extended faculties of the Analytical Engine; some of which results we think we could now hint at, had we the space; and others, which it may not yet be possible to foresee, but which would be brought forth by the daily increasing requirements of science, and by a more intimate practical acquaintance with the powers of the engine, were it in actual existence.

On general grounds, both of an *a priori* description as well as those founded on the scientific history and experience of mankind, we see strong presumptions that such would be the case. Nevertheless all will probably concur in feeling that the completion of the Difference Engine would be far preferable to the non-completion of any calculating engine at all. With whomsoever or wheresoever may rest the present causes of difficulty that apparently exist towards either the completion of the old engine, or the commencement of the new one, we trust they will not ultimately result in this generation's being acquainted with these inventions through the medium of pen, ink and paper merely; and still more do we hope, that for the honour of our country's reputation in the future pages of history, these causes will not lead to the completion of the undertaking by some *other* nation or government. This could not but be matter of just regret; and equally so, whether the obstacles may have originated in private interests and feelings, in considerations of a more public description, or in causes combining the nature of both such solutions.

We refer the reader to the "Edinburgh Review" of July, 1834, for a very able account of the Difference Engine. The writer of the article we allude to, has selected as his prominent matter for exposition, a wholly different view of the subject from that which M. Menabrea has chosen. The former chiefly treats it under its mechanical aspect, entering but slightly into the mathematical principles of which that engine is the representative, but giving, in considerable length, many details of the mechanism and contrivances by means of which it tabulates the various orders of differences. M. Menabrea, on the contrary, exclusively develops the analytical view; taking it for granted that mechanism is able to perform certain processes, but without attempting to explain *how*; and devoting his whole attention to explanations and illustrations of the manner in which analytical laws can be so arranged and combined as to bring every branch of that vast subject within the grasp of the assumed powers of mechanism. It is obvious that, in the invention of a calculating engine, these two branches of the subject are equally essential fields of investigation, and that on their mutual adjustment, one to the other, must depend all success. They must be made to meet each other, so that the weak points in the powers of either department may be compensated by the strong points in those of the other. They are indissolubly connected, though so different in their intrinsic nature that perhaps the same mind might not be likely to prove equally profound or successful in both. We know those who doubt whether the powers of mechanism will in practice prove adequate in all respects to the demands made upon them in the working of such complicated trains of machinery as those of the above engines, and who apprehend that unforeseen practical difficulties and disturbances will arise in the way of accuracy and of facility of operation. The Difference Engine, however, appears to us to be in a great measure an answer to these doubts. It is complete as far as it goes, and it does work with all the anticipated success. The Analytical Engine, far from being more complicated, will in many respects be of simpler construction; and it is a remarkable circumstance attending it, that with very *simplified* means it is so much more powerful.

The article in the "Edinburgh Review" was written some time previous to the occurrence of any ideas such as afterwards led to the invention of the Analytical Engine; and in the nature of the Difference Engine there is much less that would invite a writer to take exclusively, or even prominently, the mathematical view of it, than in that of the Analytical Engine; although mechanism has undoubtedly gone much further to meet mathematics, in the case of this engine, than of the former one. Some publication embracing the *mechanical* view of the Analytical Engine is a desideratum which we trust will be supplied before long.

Those who may have the patience to study a moderate quantity of rather dry details, will find ample compensation, after perusing the article of 1834, in the clearness with which a succinct view will have been attained of the various practical steps through which mechanism can accomplish certain processes; and they will also find themselves still further capable of appreciating M. Menabrea's more comprehensive and generalized memoir. The very difference in the style and object of these two articles, makes them peculiarly valuable to each other; at least for the purposes of those who really desire something more than a merely superficial and popular comprehension of the subject of calculating engines. A. A. L.

Note B—Page 349

That portion of the Analytical Engine here alluded to is called the storehouse. It contains an indefinite number of the columns of discs described by M. Menabrea. The reader may picture to himself a pile of rather large draughtsmen heaped perpendicularly one above another to a considerable height, each counter having the digits from 0 to 9 inscribed on its *edge* at equal intervals; and if he then conceives that the counters do not actually lie one upon another so as to be in contact, but are fixed at small intervals of vertical distance on a common axis which passes perpendicularly through their centres, and around which each disc can *revolve horizontally* so that any required digit amongst those inscribed on its margin can be brought into view, he will have a good idea of one of these columns. The *lowest* of the discs on any column belongs to the units, the next above to the tens, the next above this to the hundreds, and so on. Thus, if we wished to inscribe 1345 on a column of the engine, it would stand thus—

1
3
4
5

In the Difference Engine there are seven of these columns placed side by side in a row, and the working mechanism extends behind them; the general form of the whole mass of machinery is that of a quadrangular prism (more or less approaching to the cube); the results always appearing on that perpendicular face of the engine which contains the columns of discs, opposite to which face a spectator may place himself. In the Analytical Engine there would be many more of these columns, probably at least two hundred. The precise form and arrangement which the whole mass of its mechanism will assume is not yet finally determined.

We may conveniently represent the columns of discs on paper in a diagram like the following—

V_1	V_2	V_3	V_4	etc.
○	○	○	○	etc.
o	o	o	o	
o	o	o	o	etc.
o	o	o	o	
o	o	o	o	
□	□	□	□	etc.

The *V*'s are for the purpose of convenient reference to any column, either in writing or speaking, and are consequently numbered. The reason why the letter *V* is chosen for this purpose in preference to any other letter, is because these columns are designated (as the reader will find in proceeding with the Memoir) the *Variables*, and sometimes the *Variable columns*, or the *columns of Variables*. The origin of this appellation is, that the values on the columns are destined to change, that is to *vary* in every conceivable manner. But it is necessary to guard against the natural misapprehension that the columns are only intended to receive the values of the *variables* in an analytical formula, and not of the *constants*. The columns are called Variables on a ground wholly unconnected with the *analytical* distinction between constants and variables. In order to prevent the possibility of confusion, we have, both in the translation and in the notes, written Variable with a capital letter when we use the word to signify a *column of the engine*, and variable with a small letter when we mean the *variable of a formula*. Similarly, *Variable-cards* signify any cards that belong to a column of the engine.

To return to the explanation of the diagram: each circle at the top is intended to contain the algebraic sign + or —, either of which can be substituted* for the other, according as the number represented on the column below is positive or negative. In a similar manner any other purely *symbolical* results of algebraical processes might be made to appear in these circles. In Note A the practicability of developing *symbolical* with no less ease than *numerical* results has been touched on.

The zeros beneath the *symbolic* circles represent each of them a disc, supposed to have the digit 0 presented in front. Only four tiers of zeros have been figured in the diagram, but these may be considered as representing thirty or forty, or any number of tiers of discs that may be required. Since each disc can present any digit, and each circle any sign, the discs of every column may be so adjusted† as to express any positive or negative number whatever within the limits of the machine; which limits depend on the *perpendicular* extent of the mechanism, that is, on the number of discs to a column.

Each of the squares below the zeros is intended for the inscription of any *general* symbol or combination of symbols we please; it being understood that the number represented on the column immediately above, is the numerical value of that symbol, or combination of symbols. Let us,

* A fuller account of the manner in which the *signs* are regulated, is given in M. Menabrea's Memoir, pages 354–6. He himself expresses doubts (in a note of his own at the bottom of the latter page) as to his having been likely to hit on the precise methods really adopted; his explanation being merely a conjectural one. That it *does* accord precisely with the fact is a remarkable circumstance, and affords a convincing proof how completely M. Menabrea has been imbued with the true spirit of the invention. Indeed the whole of the above Memoir is a striking production, when we consider that M. Menabrea had had but very slight means for obtaining any adequate ideas respecting the Analytical Engine. It requires, however, a considerable acquaintance with the abstruse and complicated nature of such a subject, in order fully to appreciate the penetration of the writer who could take so just and comprehensive a view of it upon such limited opportunity.

† This adjustment is done by hand merely.

for instance, represent the three quantities a, n, x, and let us further suppose that $a = 5$, $n = 7$, $x = 98$. We should have—

V_1	V_2	V_3	V_4	etc.
+*	+	+	+	
0	0	0	0	
0	0	0	0	etc.
0	0	9	0	
5	7	8	0	etc.
a	n	x		

We may now combine these symbols in a variety of ways, so as to form any required function or functions of them, and we may then inscribe each such function below brackets, every bracket uniting together those quantities (and those only) which enter into the function inscribed below it. We must also, when we have decided on the particular function whose numerical value we desire to calculate, assign another column to the right-hand for receiving the *results*, and must inscribe the function in the square below this column. In the above instance we might have any one of the following functions—

$$ax^n,\ x^{an},\ a\,.\,n\,.\,x,\ \frac{a}{n}x,\ a + n + x, \text{ etc.}$$

Let us select the first. It would stand as follows, previous to calculation—

V_1	V_2	V_3	V_4	etc.
+	+	+	+	
0	0	0	0	etc.
0	0	0	0	
0	0	9	0	
5	7	8	0	etc.
a	n	x	ax^n	etc.
	ax^n (bracket uniting a, n, x)			

The data being given, we must now put into the engine the cards proper for directing the operations in the case of the particular function chosen. These operations would in this instance be—

First, six multiplications in order to get $x^n (= 98^7$ for the above particular data).

Secondly, one multiplication in order then to get $a\,.\,x^n\ (= 5 \times 98^7)$.

* It is convenient to omit the circles whenever the sign + or — can be actually represented.

In all, seven multiplications to complete the whole process. We may thus represent them—

$$(\times, \times, \times, \times, \times, \times, \times), \text{ or } 7(\times)$$

The multiplications would, however, at successive stages in the solution of the problem, operate on pairs of numbers, derived from *different* columns. In other words, the *same operation* would be performed on different *subjects of operation*. And here again is an illustration of the remarks made in the preceding Note on the independent manner in which the engine directs its *operations*. In determining the value of ax^n, the *operations* are *homogeneous*, but are distributed amongst different *subjects of operation*, at successive stages of the computation. It is by means of certain punched cards, belonging to the Variables themselves, that the action of the operations is so *distributed* as to suit each particular function. The *Operation-cards* merely determine the succession of operations in a general manner. They in fact throw all that portion of the mechanism included in the *mill*, into a series of different *states*, which we may call the *adding state*, or the *multiplying state*, etc., respectively. In each of these states the mechanism is ready to act in the way peculiar to that state, on any pair of numbers which may be permitted to come within its sphere of action. Only *one* of these operating states of the mill can exist at a time; and the nature of the mechanism is also such that only *one pair of numbers* can be received and acted on at a time. Now, in order to secure that the mill shall receive a constant supply of the proper pairs of numbers in succession and that it shall also rightly locate the result of an operation performed upon any pair, each Variable has cards of its own belonging to it. It has, first, a class of cards whose business it is to *allow* the number on the Variable to pass into the mill, there to be operated upon. These cards may be called the *Supplying-cards*. *They* furnish the mill with its proper food. Each Variable has, secondly, another class of cards, whose office it is to allow the Variable to *receive* a number *from* the mill. These cards may be called the *Receiving-cards*. *They* regulate the location of results, whether temporary or ultimate results. The Variable-cards in general (including both the preceding classes) might, it appears to us, be even more appropriately designated the Distributive-cards, since it is through their means that the action of the operations, and the results of this action, are rightly *distributed*.

There are *two varieties* of the *Supplying* Variable-cards, respectively adapted for fulfilling two distinct subsidiary purposes: but as these modifications do not bear upon the present subject, we shall notice them in another place.

In the above case of ax^n, the Operation-cards merely order seven multiplications, that is, they order the mill to be in the *multiplying state* seven successive times (without any reference to the particular columns whose numbers are to be acted upon). The proper Distributive

Variable-cards step in at each successive multiplication, and cause the distributions requisite for the particular case.

For x^{an}	the operations would be	$34(\times)$
„ $a \, . \, n \, . \, x$	„ „ „	$(\times, \times)$, or $2(\times)$
„ $\frac{a}{n} \, . \, x$	„ „ „	$(\div, \times)$
„ $a + n + x$	„ „ „	$(+, +)$, or $2(+)$

The engine might be made to calculate all these in succession. Having completed ax^n, the function x^{an} might be written under the brackets instead of ax^n, and a new calculation commenced (the appropriate Operation and Variable-cards for the new function of course coming into play). The results would then appear on V_5. So on for any number of different functions of the quantities a, n, x. Each *result* might either permanently remain on its column during the succeeding calculations, so that when all the functions had been computed, their values would simultaneously exist on V_4, V_5, V_6, etc.; or each result might (after being printed off, or used in any specified manner) be effaced, to make way for its successor. The square under V_4 ought, for the latter arrangement, to have the functions ax^n, x^{an}, anx, etc., successively inscribed in it.

Let us now suppose that we have *two* expressions whose values have been computed by the engine independently of each other (each having its own group of columns for data and results). Let them be ax^n, $b \, . \, p \, . \, y$. They would then stand as follows on the columns—

V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8	V_9
+	+	+	+	+	+	+	+	+
0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	
a	n	x	ax^n	b	p	y	bpy	$\frac{ax^n}{bpy}$

We may now desire to combine together these two *results*, in any manner we please; in which case it would only be necessary to have an additional card or cards, which should order the requisite operations to be performed with the numbers on the two result-columns, V_4 and V_8, and the *result of these further operations* to appear on a new column, V_9. Say that we wish to divide ax^n by $b \, . \, p \, . \, y$. The numerical value of this division would then appear on the column V_9, beneath which we have inscribed $\frac{ax^n}{bpy}$.

The whole series of operations from the beginning would be as follows (n being $= 7$)—

$$\{7(\times), 2(\times), \div\}, \text{ or } \{9(\times), \div\}$$

This example is introduced merely to show that we may, if we please, retain separately and permanently any *intermediate* results (like ax^n, $b \,.\, p \,.\, y$), which occur in the course of processes having an ulterior and more complicated result as their chief and final object $\left(\text{like } \dfrac{ax^n}{bpy}\right)$.

Any group of columns may be considered as representing a *general* function, until a *special* one has been implicitly impressed upon them through the introduction into the engine of the Operation and Variable-cards made out for a *particular* function. Thus, in the preceding example, V_1, V_2, V_3, V_5, V_6, V_7 represent the *general* function $\phi(a, n, b, p, x, y)$ until the function $\dfrac{ax^n}{b \,.\, p \,.\, y}$ has been determined on, and *implicitly* expressed by the placing of the right cards in the engine. The actual working of the mechanism, as regulated by these cards, then *explicitly* develops the value of the function. The inscription of a function under the brackets, and in the square under the result-column, in no way influences the processes or the results, and is merely a memorandum for the observer, to remind him of what is going on. It is the Operation and the Variable-cards only, which in reality determine the function. Indeed it should be distinctly kept in mind that the inscriptions within *any* of the squares, are quite independent of the mechanism or workings of the engine, and are nothing but arbitrary memorandums placed there at pleasure to assist the spectator.

The further we analyse the manner in which such an engine performs its processes and attains its results, the more we perceive how distinctly it places in a true and just light the mutual relations and connexion of the various steps of mathematical analysis, how clearly it separates those things which are in reality distinct and independent, and unites those which are mutually dependent. A. A. L.

Note C—Page 350

Those who may desire to study the principles of the Jacquard-loom in the most effectual manner, viz. that of practical observation, have only to step into the Adelaide Gallery or the Polytechnic Institution. In each of these valuable repositories of scientific *illustration*, a weaver is constantly working at a Jacquard-loom, and is ready to give any information that may be desired as to the construction and modes of acting of his apparatus. The volume on the manufacture of silk, in Lardner's Cyclopaedia, contains a chapter on the Jacquard-loom, which may also be consulted with advantage.

The mode of application of the cards, as hitherto used in the art of weaving, was not found, however, to be sufficiently powerful for all the simplifications which it was desirable to attain in such varied and complicated processes as those required in order to fulfil the purposes of an Analytical Engine. A method was devised of what was technically designated *backing* the cards in certain groups according to certain laws. The object of this extension is to secure the possibility of bringing any particular card or set of cards into use *any number of times successively* in the solution of one problem. Whether this power shall be taken advantage of or not, in each particular instance, will depend on the nature of the operations which the problem under consideration may require. The process is alluded to by M. Menabrea in page 353, and it is a very important simplification. It has been proposed to use it for the reciprocal benefit of that art, which, while it has itself no apparent connexion with the domains of abstract science, has yet proved so valuable to the latter, in suggesting the principles which, in their new and singular field of application, seem likely to place *algebraical* combinations not less completely within the province of mechanism, than are all those varied intricacies of which *intersecting threads* are susceptible. By the introduction of the system of *backing* into the Jacquard-loom itself, patterns which should possess symmetry, and follow regular laws of any extent, might be woven by means of comparatively few cards.

Those who understand the mechanism of this loom will perceive that the above improvement is easily effected in practice, by causing the prism over which the train of pattern-cards is suspended, to revolve *backwards* instead of *forwards*, at pleasure, under the requisite circumstances; until, by so doing, any particular card, or set of cards, that has done duty once, and passed on in the ordinary regular succession, is brought back to the position it occupied just before it was used the preceding time. The prism then resumes its *forward* rotation, and thus brings the card or set of cards in question into play a second time. This process may obviously be repeated any number of times. A. A. L.

Note D—Page 353

We have represented the solution of these two equations, with every detail, in a diagram* similar to those used in Note B; but additional explanations are requisite, partly in order to make this more complicated case perfectly clear, and partly for the comprehension of certain indications and notations not used in the preceding diagrams. Those who may wish to understand Note G completely, are recommended to pay particular attention to the contents of the present Note, or they will not otherwise comprehend the similar notation and indications when applied to a much more complicated case.

* See the diagram on page 384.

In all calculations, the columns of Variables used may be divided into three classes—

1. Those on which the data are inscribed.

2. Those intended to receive the final results.

3. Those intended to receive such intermediate and temporary combinations of the primitive data as are not to be permanently retained, but are merely needed for *working with*, in order to attain the ultimate results. Combinations of this kind might properly be called *secondary data*. They are in fact so many *successive stages* towards the final result. The columns which receive them are rightly named *Working-Variables*, for their office is in its nature purely *subsidiary* to other purposes. They develop an intermediate and transient class of results, which unite the original data with the final results.

The Result-Variables sometimes partake of the nature of Working-Variables. It frequently happens that a Variable destined to receive a final result is the recipient of one or more intermediate values successively, in the course of the processes. Similarly, the Variables for data often become Working-Variables, or Result-Variables, or even both in succession. It so happens, however, that in the case of the present equations the three sets of offices remain throughout perfectly separate and independent.

It will be observed, that in the squares below the *Working*-Variables nothing is inscribed. Any one of these Variables is in many cases destined to pass through various values successively during the performance of a calculation (although in these particular equations no instance of this occurs). Consequently no *one fixed* symbol, or combination of symbols, should be considered as properly belonging to a merely *Working*-Variable; and as a general rule their squares are left blank. Of course in this, as in all other cases where we mention a *general* rule, it is understood that many particular exceptions may be expedient.

In order that all the indications contained in the diagram may be completely understood, we shall now explain two or three points, not hitherto touched on. When the value on any Variable is called into use, one of two consequences may be made to result. Either the value may *return* to the Variable after it has been used, in which case it is ready for a second use if needed; or the Variable may be made zero. (We are of course not considering a third case, of not unfrequent occurrence, in which the same Variable is destined to receive the *result* of the very operation which it has just supplied with a number.) Now the ordinary rule is, that the value *returns* to the Variable; unless it has been foreseen that no use for that value can recur, in which case zero is substituted. At the *end* of a calculation, therefore, every column ought as a general rule to be zero, excepting those for results. Thus it will be seen by the diagram, that when m, the value on V_0, is used for the second time by Operation 5, V_0 becomes 0, since m is not again needed; that similarly, when

$(mn' - m'n)$, on V_{12}, is used for the third time by Operation 11, V_{12} becomes zero, since $(mn' - m'n)$ is not again needed. In order to provide for the one or the other of the courses above indicated, there are *two* varieties of the *Supplying* Variable-cards. One of these varieties has provisions which cause the number given off from any Variable to *return* to that Variable after doing its duty in the mill. The other variety has provisions which cause *zero* to be substituted on the Variable, for the number given off. These two varieties are distinguished, when needful, by the respective appellations of the *Retaining* Supply-cards and the *Zero* Supply-cards. We see that the *primary* office (see Note B) of both these varieties of cards is the same; they only differ in their *secondary* office.

Every Variable thus has belonging to it *one* class of *Receiving* Variable-cards and *two* classes of *Supplying* Variable-cards. It is plain, however, that only the *one* or the *other* of these two latter classes can be used by any one Variable for *one* operation; never *both* simultaneously; their respective functions being mutually incompatible.

It should be understood that the Variable-cards are not placed in *immediate contiguity* with the columns. Each card is connected by means of wires with the column it is intended to act upon.

Our diagram ought in reality to be placed side by side with M. Menabrea's corresponding table, so as to be compared with it, line for line belonging to each operation. But it was unfortunately inconvenient to print them in this desirable form. The diagram is, in the main, merely another manner of indicating the various relations denoted in M. Menabrea's table. Each mode has some advantages and some disadvantages. Combined, they form a complete and accurate method of registering every step and sequence in all calculations performed by the engine.

No notice has yet been taken of the *upper* indices which are added to the left of each V in the diagram; an addition which we have also taken the liberty of making to the V's in M. Menabrea's tables of pages 354, 357, since it does not *alter* anything therein represented by him, but merely *adds* something to the previous indications of those tables. The *lower* indices are obviously indices of *locality* only, and are wholly independent of the operations performed or of the results obtained, their value continuing unchanged during the performance of calculations. The *upper* indices, however, are of a different nature. Their office is to indicate any *alteration* in the value which a Variable represents; and they are of course liable to changes during the processes of a calculation. Whenever a Variable has only zeros upon it, it is called ${}^0V$; the moment a value appears on it (whether that value be placed there arbitrarily, or appears in the natural course of a calculation), it becomes ${}^1V$. If this value gives place to another value, the Variable becomes ${}^2V$, and so forth. Whenever a *value* again gives place to *zero*, the Variable again becomes ${}^0V$, even if it had been ${}^nV$ the moment before. If a *value* then again be substituted, the

Variable becomes ${}^{n+1}V$ (as it would have done if it had not passed through the intermediate ${}^{0}V$); etc., etc. Just before any calculation is commenced, and after the data have been given, and everything adjusted and prepared for setting the mechanism in action, the upper indices of the Variables for data are all unity, and those for the Working- and Result-Variables are all zero. In this state the diagram represents them.*

There are several advantages in having a set of indices of this nature; but these advantages are perhaps hardly of a kind to be immediately perceived, unless by a mind somewhat accustomed to trace the successive steps by means of which the engine accomplishes its purposes. We have only space to mention in a general way, that the whole notation of the tables is made more consistent by these indices, for they are able to mark a *difference* in certain cases, where there would otherwise be an apparent *identity* confusing in its tendency. In such a case as $V_n = V_p + V_n$ there is more clearness and more consistency with the usual laws of algebraical notation, in being able to write ${}^{m+1}V_n = {}^{q}V_p + {}^{m}V_n$. It is also obvious that the indices furnish a powerful means of tracing back the derivation of any result; and of registering various circumstances concerning that *series of successive substitutions*, of which every *result* is in fact merely the final consequence; circumstances that may in certain cases involve relations which it is important to observe, either for purely analytical reasons, or for practically adapting the workings of the engine to their occurrence. The series of substitutions which lead to the equations of the diagram are as follows—

$$\overset{(1)}{{}^{1}V_{16}{}^{*}} = \overset{(2)}{\frac{{}^{1}V_{14}}{{}^{1}V_{12}}} = \overset{(3)}{\frac{{}^{1}V_{10} - {}^{1}V_{11}}{{}^{1}V_{6} - {}^{1}V_{7}}} = \overset{(4)}{\frac{{}^{1}V_{0} \cdot {}^{1}V_{5} - {}^{1}V_{2} \cdot {}^{1}V_{3}}{{}^{1}V_{0} \cdot {}^{1}V_{4} - {}^{1}V_{3} \cdot {}^{1}V_{1}}} = \frac{d'm - dm'}{mn' - m'n}$$

$$\overset{(1)}{{}^{1}V_{15}} = \overset{(2)}{\frac{{}^{1}V_{13}}{{}^{1}V_{12}}} = \overset{(3)}{\frac{{}^{1}V_{8} - {}^{1}V_{9}}{{}^{1}V_{6} - {}^{1}V_{7}}} = \overset{(4)}{\frac{{}^{1}V_{2} \cdot {}^{1}V_{4} - {}^{1}V_{5} \cdot {}^{1}V_{1}}{{}^{1}V_{0} \cdot {}^{1}V_{4} - {}^{1}V_{3} \cdot {}^{1}V_{1}}} = \frac{dn' - d'n}{mn' - m'n}$$

There are *three* successive substitutions for each of these equations. The formulae (2), (3), and (4) are *implicitly* contained in (1), which latter we may consider as being in fact the *condensed* expression of any of the former. It will be observed that every succeeding substitution must contain *twice* as many V's as its predecessor. So that if a problem requires n substitutions, the successive series of numbers for the V's in the whole of them will be 2, 4, 8, 16, . . . 2^n.

The substitutions in the preceding equations happen to be of little value towards illustrating the power and uses of the upper indices; for

* We recommend the reader to trace the successive substitutions backwards from (1) to (4), in M. Menabrea's Table. This he will easily do by means of the upper and lower indices, and it is interesting to observe how each V successively ramifies (so to speak) into two other V's in some other column of the table; until at length the V's of the original data are arrived at.

DIAGRAM BELONGING TO NOTE D

Number of Operations	Nature of Operations	Variables for Data						Working Variables									Variables for Results	
		$^1V_0$	$^1V_1$	$^1V_2$	$^1V_3$	$^1V_4$	$^1V_5$	$^0V_6$	$^0V_7$	$^0V_8$	$^0V_9$	$^0V_{10}$	$^0V_{11}$	$^0V_{12}$	$^0V_{13}$	$^0V_{14}$	$^0V_{15}$	$^0V_{16}$
		+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		m	n	d	m'	n'	d'										$\frac{dn'-d'n}{mn'-m'n}=x$	$\frac{d'm-dm'}{mn'-m'n}=y$
1	×	m	—	—	—	n'	—	mn'										
2	×	—	n	—	m'	—	—	—	$m'n$									
3	×	—	—	d	—	0	—	—	—	dn'								
4	×	—	0	—	—	—	d'	—	—	—	$d'n$							
5	×	0	—	—	—	—	0	—	—	—	—	$d'm$						
6	×	—	—	0	0	—	—	—	—	—	—	—	dm'					
7	−	—	—	—	—	—	—	0	0	—	—	—	—	$(mn'-m'n)$				
8	−	—	—	—	—	—	—	—	—	0	0	—	—	—	$(dn'-d'n)$			
9	−	—	—	—	—	—	—	—	—	—	—	0	0	—	—	$(d'm-dm')$		
10	÷	—	—	—	—	—	—	—	—	—	—	—	—	$(mn'-m'n)$	0	—	$\frac{dn'-d'n}{mn'-m'n}=x$	
11	÷	—	—	—	—	—	—	—	—	—	—	—	—	0	—	0	—	$\frac{d'm-dm'}{mn'-m'n}=y$

owing to the nature of these particular equations the indices are all unity throughout. We wish we had space to enter more fully into the relations which these indices would in many cases enable us to trace.

M. Menabrea incloses the three centre columns of his table under the general title *Variable-cards.* The *V*'s, however, in reality all represent the actual *Variable-columns* of the engine, and not the cards that belong to them. Still the title is a very just one, since it is through the special action of certain Variable-cards (when *combined* with the more generalized agency of the Operation-cards) that every one of the particular relations he has indicated under that title is brought about.

Suppose we wish to ascertain how often any *one* quantity, or combination of quantities, is brought into use during a calculation. We easily ascertain *this*, from the inspection of any vertical column or columns of the diagram in which that quantity may appear. Thus, in the present case, we see that all the data, and all the intermediate results likewise, are used twice, excepting $(mn' - m'n)$, which is used three times.

The *order* in which it is possible to perform the operations for the present example, enables us to effect all the eleven operations of which it consists, with only *three Operation-cards*; because the problem is of such a nature that it admits of each *class* of operations being performed in a group together; all the multiplications one after another, all the subtractions one after another, etc. The operations are $\{6(\times), 3(-), 2(\div)\}$.

Since the very definition of an operation implies that there must be *two* numbers to act upon, there are of course *two Supplying* Variable-cards necessarily brought into action for every operation, in order to furnish the two proper numbers. (See Note B.) Also, since every operation must produce a *result*, which must be placed *somewhere*, each operation entails the action of a *Receiving* Variable-card, to indicate the proper locality for the result. Therefore, at least three times as many Variable-cards as there are *operations* (not *Operation-cards*, for these, as we have just seen, are by no means always as numerous as the *operations*) are brought into use in every calculation. Indeed, under certain contingencies, a still larger proportion is requisite; such, for example, would probably be the case when the same result has to appear on more than one Variable simultaneously (which is not unfrequently a provision necessary for subsequent purposes in a calculation), and in some other cases which we shall not here specify. We see therefore that a great disproportion exists between the amount of *Variable* and of *Operation*-cards requisite for the working of even the simplest calculation.

All calculations do not admit, like this one, of the operations of the same nature being performed in groups together. Probably very few do so without exceptions occurring in one or other stage of the progress; and some would not admit it at all. The *order* in which the operations shall be performed in every particular case, is a very interesting and curious question, on which our space does not permit us fully to enter. In almost every computation a great *variety* of arrangements for the succession of the processes is possible, and various considerations must influence the selection amongst them for the purposes of a Calculating Engine. One essential object is to choose that arrangement which shall tend to reduce to a minimum the *time* necessary for completing the calculation.

It must be evident how multifarious and how mutually complicated are the considerations which the workings of such an engine involve. There are frequently several distinct *sets of effects* going on simultaneously; all in a manner independent of each other, and yet to a greater or less degree exercising a mutual influence. To adjust each to every other, and indeed even to perceive and trace them out with perfect correctness and success, entails difficulties whose nature partakes to a certain extent of those involved in every question where *conditions* are very numerous and inter-complicated; such as for instance the estimation of the mutual relations amongst *statistical* phenomena, and of those involved in many other classes of facts. A. A. L.

Note E—Page 356

This example has evidently been chosen on account of its brevity and simplicity, with a view merely to explain the *manner* in which the engine would proceed in the case of an *analytical calculation containing variables*, rather than to illustrate the *extent of its powers* to solve cases of a difficult and complex nature. The equations of page 352 are in fact a more complicated problem than the present one.

We have not subjoined any diagram of its development for this new example, as we did for the former one, because this is unnecessary after the full application already made of those diagrams to the illustration of M. Menabrea's excellent tables.

It may be remarked that a slight discrepancy exists between the formulae

$$(a + bx^1)$$

$$(A + B\cos^1 x)$$

given in the Memoir as the *data* for calculation, and the *results* of the calculation as developed in the last division of the table which accompanies it. To agree perfectly with this latter, the data should have been given as

$$(ax^0 + bx^1)$$

$$(A\cos^0 x + B\cos^1 x)$$

The following is a more complicated example of the manner in which the engine would compute a trigonometrical function containing variables.

To multiply

$$A + A_1 \cos \theta + A_2 \cos 2\theta + A_3 \cos 3\theta + \ldots$$

by

$$B + B_1 \cos \theta$$

Let the resulting products be represented under the general form

$$C_0 + C_1 \cos \theta + C_2 \cos 2\theta + C_3 \cos 3\theta + \quad . \quad . \quad (1)$$

This trigonometrical series is not only in itself very appropriate for illustrating the processes of the engine, but is likewise of much practical interest from its frequent use in astronomical computations. Before proceeding further with it, we shall point out that there are three very distinct classes of ways in which it may be desired to deduce numerical values from any analytical formula.

First, we may wish to find the collective numerical value of the *whole formula*, without any reference to the quantities of which that formula is a function, or to the particular mode of their combination and distribution, of which the formula is the result and representative. Values of this kind are of a strictly arithmetical nature in the most limited sense of the term, and retain no trace whatever of the processes through which they have been deduced. In fact, any one such numerical value may have been attained from an *infinite variety* of data, or of problems. The values for x and y in the two equations (see Note D), come under this class of numerical results.

Secondly, we may propose to compute the collective numerical value of *each term* of a formula, or of a series, and to keep these results separate. The engine must in such a case appropriate as many columns to *results* as there are terms to compute.

Thirdly, it may be desired to compute the numerical value of various *subdivisions of each term*, and to keep all these results separate. It may be required, for instance, to compute each coefficient separately from its variable, in which particular case the engine must appropriate *two* result-columns to *every term that contains both a variable and coefficient*.

There are many ways in which it may be desired in special cases to distribute and keep separate the numerical values of different parts of an algebraical formula; and the power of effecting such distributions to any extent is essential to the *algebraical* character of the Analytical Engine. Many persons who are not conversant with mathematical studies, imagine that because the business of the engine is to give its results in *numerical notation*, the *nature of its processes* must consequently be *arithmetical* and *numerical*, rather than *algebraical* and *analytical*. This is an error. The engine can arrange and combine its numerical quantities exactly as if they were *letters* or any other *general* symbols; and in fact it might bring out its results in algebraical *notation*, were provisions made accordingly. It might develop three sets of results simultaneously, viz. *symbolic* results (as already alluded to in Notes A and B); *numerical* results (its chief and primary object); and *algebraical* results in *literal* notation. This latter, however, has not been deemed a necessary or desirable addition to its powers, partly because the necessary arrangements for effecting it would increase the complexity and extent of the mechanism to a degree that would not be commensurate with the advantages, where the main object of the invention is to translate into *numerical* language general formulae of analysis already known to us, or whose laws of formation are known to us. But it would be a mistake to suppose that because its *results* are given in the *notation* of a more restricted science, its *processes* are therefore restricted to those of that science. The object of the engine is in fact to give the *utmost practical efficiency* to the resources of *numerical interpretations* of the higher science of analysis, while it uses the processes and combinations of this latter.

To return to the trigonometrical series. We shall only consider the four first terms of the factor $(A + A_1 \cos\theta + \text{etc.})$, since this will be sufficient to show the method. We propose to obtain separately the numerical value of *each coefficient* C_0, C_1, etc., of (1). The direct multiplication of the two factors gives—

$$\left.\begin{array}{l} BA + BA_1 \cos\theta + BA_2 \cos 2\theta + BA_3 \cos 3\theta + \ldots \\ \quad + B_1A \cos\theta + B_1A_1 \cos\theta \,.\, \cos\theta + B_1A_2 \cos 2\theta \,.\, \cos\theta \\ \qquad + B_1A_3 \cos 3\theta \,.\, \cos\theta \end{array}\right\} \quad (2)$$

a result which would stand thus on the engine—

Variables for Data

V_0	V_1	V_2	V_3	. . . . . .	V_{10}	V_{11}
A	A_1	A_2	A_3		B	B_1
	$\cos\theta$	$\cos 2\theta$	$\cos 3\theta$			$\cos\theta$

Variables for Results

V_{20}	V_{21}	V_{22}	V_{23}	. . . .	V_{31}	V_{32}	V_{33}	V_{34}
BA	BA_1	BA_2	BA_3		B_1A	B_1A_1	B_1A_2	B_1A_3
	$\cos\theta$	$\cos 2\theta$	$\cos 3\theta$		$\cos\theta$	$(\cos\theta \,.\, \cos\theta)$	: $(\cos 2\theta \,.\, \cos\theta)$	$(\cos 3\theta \,.\, \cos\theta)$

The variable belonging to each coefficient is written below it, as we have done in the diagram, by way of memorandum. The only further reduction which is at first apparently possible in the preceding result, would be the addition of V_{21} to V_{31} (in which case B_1A should be effaced from V_{31}). The whole operations from the beginning would then be—

First Series of Operations	Second Series of Operations	Third Series, which contains only one (final) Operation
${}^1V_{10} \times {}^1V_0 = {}^1V_{20}$	${}^1V_{11} \times {}^1V_0 = {}^1V_{31}$	${}^1V_{21} + {}^1V_{31} = {}^2V_{21}$, and
${}^1V_{10} \times {}^1V_1 = {}^1V_{21}$	${}^1V_{11} \times {}^1V_1 = {}^1V_{32}$	V_{31} becomes $= 0$
${}^1V_{10} \times {}^1V_2 = {}^1V_{22}$	${}^1V_{11} \times {}^1V_2 = {}^1V_{33}$	
${}^1V_{10} \times {}^1V_3 = {}^1V_{23}$	${}^1V_{11} \times {}^1V_3 = {}^1V_{34}$	

We do not enter into the same detail of *every* step of the processes as in the examples of Notes D and G, thinking it unnecessary and tedious to do so. The reader will remember the meaning and use of the upper and lower indices, etc., as before explained.

To proceed: we know that

$$\cos n\theta \,.\, \cos\theta = \tfrac{1}{2}\cos\overline{n+1}\,.\,\theta + \tfrac{1}{2}\cos\overline{n-1}\,.\,\theta \qquad . \qquad (3)$$

Consequently, a slight examination of the second line of (2) will show that by making the proper substitutions, (2) will become

BA	$+ BA_1 \,.\, \cos\theta$	$+ BA_2 \,.\, \cos 2\theta$	$+ BA_3 \,.\, \cos 3\theta$	
	$+ B_1A \,.\, \cos\theta$			
$+ \frac{1}{2}B_1A_1$		$+ \frac{1}{2}B_1A_1 \,.\, \cos 2\theta$		
	$+ \frac{1}{2}B_1A_2 \,.\, \cos\theta$		$+ \frac{1}{2}B_1A_2 \,.\, \cos 3\theta$	
		$+ \frac{1}{2}B_1A_3 \,.\, \cos 2\theta$		$+ \frac{1}{2}B_1A_3 \,.\, \cos 4\theta$
C_0	C_1	C_2	C_3	C_4

These coefficients should respectively appear on

V_{20} $\quad V_{21}$ $\quad V_{22}$ $\quad V_{23}$ $\quad V_{24}$

We shall perceive, if we inspect the particular arrangement of the results in (2) on the Result-columns as represented in the diagram, that, in order to effect this transformation, each successive coefficient upon V_{32}, V_{33}, etc. (beginning with V_{32}), must through means of proper cards be divided by *two*;* and that one of the halves thus obtained must be added to the coefficient on the Variable which precedes it by ten columns, and the other half to the coefficient on the Variable which precedes it by twelve columns; V_{32}, V_{33}, etc., themselves becoming zeros during the process.

* This division would be managed by ordering the number two to appear on any separate new column which should be conveniently situated for the purpose, and then directing this column (which is in the strictest sense a *Working*-Variable) to divide itself successively with V_{32}, V_{33}, etc.

This series of operations may be thus expressed—

Fourth Series

$$\begin{cases} {}^1V_{32} \div 2 + {}^1V_{22} = {}^2V_{22} = BA_2 + \tfrac{1}{2}B_1A_1 \\ {}^1V_{32} \div 2 + {}^1V_{20} = {}^2V_{20} = BA + \tfrac{1}{2}B_1A_1 & = C_0 \end{cases}$$

$$\begin{cases} {}^1V_{33} \div 2 + {}^1V_{23} = {}^2V_{23} = BA_3 + \tfrac{1}{2}B_1A_2 & = C_3{}^* \\ {}^1V_{33} \div 2 + {}^2V_{21} = {}^3V_{21} = BA_1 + B_1A + \tfrac{1}{2}B_1A_2 & = C_1 \end{cases}$$

$$\begin{cases} {}^1V_{34} \div 2 + {}^0V_{24} = {}^1V_{24} = \tfrac{1}{2}B_1A_3 & = C_4 \\ {}^1V_{34} \div 2 + {}^2V_{22} = {}^3V_{22} = BA_2 + \tfrac{1}{2}B_1A_1 + \tfrac{1}{2}B_1A_3 & = C_2 \end{cases}$$

The calculation of the coefficients C_0, C_1, etc., of (1), would now be completed, and they would stand ranged in order on V_{20}, V_{21}, etc. It will be remarked, that from the moment the fourth series of operations is ordered, the Variables V_{31}, V_{32}, etc., cease to be *Result*-Variables, and become mere *Working*-Variables.

The substitution made by the engine of the processes in the second side of (3) for those in the first side, is an excellent illustration of the manner in which we may arbitrarily order it to substitute any function, number or process, at pleasure, for any other function, number or process, on the occurrence of a specified contingency.

We will now suppose that we desire to go a step further, and to obtain the numerical value of each *complete* term of the product (1), that is of each *coefficient* and *variable united*, which for the $(n + 1)$th term would be $C_n \,.\, \cos n\theta$.

We must for this purpose place the variables themselves on another set of columns, V_{41}, V_{42}, etc., and then order their successive multiplication by V_{21}, V_{22}, etc., each for each. There would thus be a final series of operations as follows—

Fifth and Final Series of Operations

$$\begin{aligned} {}^2V_{20} \times {}^0V_{40} &= {}^1V_{40} \\ {}^3V_{21} \times {}^0V_{41} &= {}^1V_{41} \\ {}^3V_{22} \times {}^0V_{42} &= {}^1V_{42} \\ {}^2V_{23} \times {}^0V_{43} &= {}^1V_{43} \\ {}^1V_{24} \times {}^0V_{44} &= {}^1V_{44} \end{aligned}$$

(N.B. that V_{40} being intended to receive the coefficient on V_{20} which has *no* variable, will only have $\cos 0\,\theta (= 1)$ inscribed on it, preparatory to commencing the fifth series of operations.)

* It should be observed, that were the rest of the factor $(A + A_1 \cos\theta + \text{etc.})$ taken into account, instead of *four* terms only, C_3 would have the additional term $\tfrac{1}{2}B_1A_4$; and C_4 the two additional terms, BA_4, $\tfrac{1}{2}B_1A_5$. This would indeed have been the case had even *six* terms been multiplied.

From the moment that the fifth and final series of operations is ordered, the Variables V_{20}, V_{21}, etc., then in their turn cease to be *Result*-Variables and become mere *Working*-Variables; V_{40}, V_{41}, etc., being now the recipients of the ultimate results.

We should observe, that if the variables cos θ, cos 2θ, cos 3θ, etc., are furnished, they would be placed directly upon V_{41}, V_{42}, etc., like any other data. If not, a separate computation might be entered upon in a separate part of the engine, in order to calculate them, and place them on V_{41}, etc.

We have now explained how the engine might compute (1) in the most direct manner, supposing we knew nothing about the *general* term of the resulting series. But the engine would in reality set to work very differently, whenever (as in this case) we *do* know the law for the general term.

The two first terms of (1) are

$$(BA + \tfrac{1}{2}B_1A_1) + (\overline{BA_1 + B_1A + \tfrac{1}{2}B_1A_2} \,.\, \cos\theta) \quad . \quad (4)$$

and the general term for all after these is

$$(BA_n + \tfrac{1}{2}B_1 \,.\, \overline{A_{n-1} + A_{n+2}}) \cos n\theta \quad . \quad . \quad . \quad (5)$$

which is the coefficient of the $(n + 1)$th term. The engine would calculate the two first terms by means of a separate set of suitable Operation-cards, and would then need another set for the third term; which last set of Operation-cards would calculate all the succeeding terms *ad infinitum*; merely requiring certain new Variable-cards for each term to direct the operations to act on the proper columns. The following would be the successive sets of operations for computing the coefficients of $n + 2$ terms—

$$(\times, \times, \div, +), (\times, \times, \times, \div, +, +), n(\times, +, \times, \div, +)$$

Or we might represent them as follows, according to the numerical order of the operations—

$$(1, 2, \ldots 4), (5, 6, \ldots 10), n(11, 12, \ldots 15)$$

The brackets, it should be understood, point out the relation in which the operations may be *grouped*, while the comma marks *succession*. The symbol + might be used for this latter purpose, but this would be liable to produce confusion, as + is also necessarily used to represent one class of the actual operations which are the subject of that succession. In accordance with this meaning attached to the comma, care must be taken when any one group of operations recurs more than once, as is represented above by $n(11 \ldots 15)$, not to insert a comma after the number or letter prefixed to that group. n, $(11 \ldots 15)$ would stand for *an operation n, followed by the group of operations* $(11 \ldots 15)$; instead of denoting *the number of groups which are to follow each other.*

Wherever a *general term* exists, there will be a *recurring group* of operations, as in the above example. Both for brevity and for distinctness, a *recurring* *group* is called a *cycle*. A *cycle* of operations, then, must be understood to signify any *set of operations* which is repeated *more than once*. It is equally a *cycle*, whether it be repeated *twice* only, or an indefinite number of times; for it is the fact of a *repetition occurring at all* that constitutes it such. In many cases of analysis there is a *recurring group* of one or more *cycles*; that is, a *cycle of a cycle*, or a *cycle of cycles*. For instance: suppose we wish to divide a series by a series,

$$(1) \qquad \frac{a + bx + cx^2 + \ldots}{a' + b'x + c'x^2 + \ldots}$$

it being required that the result shall be developed, like the dividend and the divisor, in successive powers of x. A little consideration of (1), and of the steps through which algebraical division is effected, will show that (if the denominator be supposed to consist of p terms) the first partial quotient will be completed by the following operations—

$$(2) \qquad \{(\div), p(\times, -)\} \text{ or } \{(1), p(2, 3)\}$$

that the second partial quotient will be completed by an exactly similar set of operations, which acts on the remainder obtained by the first set, instead of on the original dividend. The whole of the processes therefore that have been gone through, by the time the *second* partial quotient has been obtained, will be—

$$(3) \qquad 2\{(\div), p(\times, -)\} \text{ or } 2\{(1), p(2, 3)\}$$

which is a cycle that includes a cycle, or a cycle of the second order. The operations for the *complete* division, supposing we propose to obtain n terms of the series constituting the quotient, will be—

$$(4) \qquad n\{(\div), p(\times, -)\} \text{ or } n\{(1), p(2, 3)\}$$

It is of course to be remembered that the process of algebraical division in reality continues *ad infinitum*, except in the few exceptional cases which admit of an exact quotient being obtained. The number n in the formula (4), is always that of the number of terms we propose to ourselves to obtain; and the nth partial quotient is the coefficient of the $(n - 1)$th power of x.

There are some cases which entail *cycles of cycles of cycles*, to an indefinite extent. Such cases are usually very complicated, and they are of extreme interest when considered with reference to the engine. The algebraical development in a series, of the nth function of any given function, is of this nature. Let it be proposed to obtain the nth function of

$$(5) \qquad \phi(a, b, c, \ldots x), \; x \text{ being the variable}$$

We should premise that we suppose the reader to understand what is meant by an nth function. We suppose him likewise to comprehend distinctly the difference between developing *an* nth *function algebraically*, and merely *calculating an* nth *function arithmetically*. If he does not, the following will be by no means very intelligible; but we have not space to give any preliminary explanations. To proceed: the law, according to which the successive functions of (5) are to be developed, must of course first be fixed on. This law may be of very various kinds. We may propose to obtain our results in successive *powers* of x, in which case the general form would be

$$C + C_1x + C_2x^2 + \text{etc.}$$

or in successive powers of n itself, the index of the function we are ultimately to obtain, in which case the general form would be

$$C + C_1n + C_2n^2 + \text{etc.}$$

and x would only enter in the coefficients. Again, other functions of x or of n instead of *powers*, might be selected. It might be in addition proposed, that the coefficients themselves should be arranged according to given functions of a certain quantity. Another mode would be to make equations arbitrarily amongst the coefficients only, in which case the several functions, according to either of which it might be possible to develop the nth function of (5), would have to be determined from the combined consideration of these equations and of (5) itself.

The *algebraical* nature of the engine (so strongly insisted on in a previous part of this Note) would enable it to follow out any of these various modes indifferently; just as we recently showed that it can distribute and separate the numerical results of any one prescribed series of processes, in a perfectly arbitrary manner. Were it otherwise, the engine could merely *compute the arithmetical* nth *function*, a result which, like any other purely arithmetical results, would be simply a collective number, bearing no traces of the data or the processes which had led to it.

Secondly, the *law* of development for the nth function being selected, the next step would obviously be to develop (5) itself, according to this law. This result would be the first function, and would be obtained by a determinate series of processes. These in most cases would include amongst them one or more *cycles* of operations.

The third step (which would consist of the various processes necessary for effecting the actual substitution of the series constituting the *first function*, for the *variable* itself) might proceed in either of two ways. It might make the substitution either wherever x occurs in the original (5), or it might similarly make it wherever x occurs in the first function itself which is the equivalent of (5). In some cases the former mode might be best, and in others the latter.

Whichever is adopted, it must be understood that the result is to appear arranged in a series following the law originally prescribed for the development of the nth function. This result constitutes the second function; with which we are to proceed exactly as we did with the first function, in order to obtain the third function; and so on, $n - 1$ times, to obtain the nth function. We easily perceive that since every successive function is arranged in a series *following the same law*, there would (after the *first* function is obtained) be a *cycle*, *of a cycle*, etc., of operations,* one, two, three, up to $n - 1$ times in order to get the nth function. We say, *after the first function is obtained*, because (for reasons on which we cannot here enter) the *first* function might in many cases be developed through a set of processes peculiar to itself, and not recurring for the remaining functions.

We have given but a very slight sketch, of the principal *general* steps which would be requisite for obtaining an nth function of such a formula as (5). The question is so exceedingly complicated, that perhaps few persons can be expected to follow, to their own satisfaction, so brief and general a statement as we are here restricted to on this subject. Still it is a very important case as regards the engine, and suggests ideas peculiar to itself, which we should regret to pass wholly without allusion. Nothing could be more interesting than to follow out, in every detail, the solution by the engine of such a case as the above; but the time, space and labour this would necessitate, could only suit a very extensive work.

To return to the subject of *cycles* of operations: some of the notation of the integral calculus lends itself very aptly to express them: (2) might be thus written—

$$(6) \qquad (\div), \sum(+1)^p (\times, -) \text{ or } (1), \sum(+1)^p (2, 3)$$

where p stands for the variable; $(+1)^p$ for the function of the variable, that is, for ϕp; and the limits are from 1 to p, or from 0 to $p - 1$, each increment being equal to unity. Similarly, (4) would be—

$$(7) \qquad \sum(+1)^n \{(\div), \sum(+1)^p (\times, -)\}$$

the limits of n being from 1 to n, or from 0 to $n - 1$,

$$(8) \qquad \text{or } \sum(+1)^n \{(1), \sum(+1)^p (2, 3)\}$$

Perhaps it may be thought that this notation is merely a circuitous way of expressing what was more simply and as effectually expressed before; and, in the above example, there may be some truth in this. But there is another description of cycles which *can* only effectually be expressed, in a condensed form, by the preceding notation. We shall call them *varying cycles*. They are of frequent occurrence, and include successive cycles of operations of the following nature—

$$(9) \qquad p(1, 2, \ldots m), \overline{p-1}(1, 2, \ldots m), \overline{p-2}(1, 2, \ldots m) \ldots \overline{p-n}(1, 2, \ldots m)$$

* A cycle that includes n other cycles, successively *contained one within another*, is called a cycle of the $n + 1$th order. A cycle may simply *include* many other cycles, and yet only be of the second order. If a series follows a certain law for a certain number of terms, and then another law for another number of terms, there will be a cycle of operations for every new law; but these cycles will not be *contained one within another*—they merely *follow each other*. Therefore their number may be infinite without influencing the *order* of a cycle that includes a repetition of such a series.

where each cycle contains the same group of operations, but in which the number of repetitions of the group varies according to a fixed rate, with every cycle. (9) can be well expressed as follows—

(10) $\sum p(1, 2, \ldots m)$, the limits of p being from $p - n$ to p

Independent of the intrinsic advantages which we thus perceive to result in certain cases from this use of the notation of the integral calculus, there are likewise considerations which make it interesting, from the connexions and relations involved in this new application. It has been observed in some of the former Notes, that the processes used in analysis form a logical system of much higher generality than the applications to number merely. Thus, when we read over any algebraical formula, considering it exclusively with reference to the processes of the engine, and putting aside for the moment its abstract signification as to the relations of quantity, the symbols $+$, $\times$, etc., in reality represent (as their immediate and proximate effect, when the formula is applied to the engine) that a certain prism which is a part of the mechanism (see Note C), turns a new face, and thus presents a new card to act on the bundles of levers of the engine; the new card being perforated with holes, which are arranged according to the peculiarities of the operation of addition, or of multiplication, etc. Again, the *numbers* in the preceding formula (8), each of them really represents one of these very pieces of card that are hung over the prism.

Now in the use made in the formulae (7), (8) and (10), of the notation of the integral calculus, we have glimpses of a similar new application of the language of the *higher* mathematics. Σ, in reality, here indicates that when a certain number of cards have acted in succession, the prism over which they revolve must *rotate backwards*, so as to bring those cards into their former position; and the limits 1 to n, 1 to p, etc., regulate how often this backward rotation is to be repeated. A. A. L.

Note F—Page 360

There is in existence a beautiful woven portrait of Jacquard, in the fabrication of which 24,000 cards were required.

The power of *repeating* the cards, alluded to by M. Menabrea in page 353, and more fully explained in Note C, reduces to an immense extent the number of cards required. It is obvious that this mechanical improvement is especially applicable wherever *cycles* occur in the mathematical operations, and that, in preparing data for calculations by the engine, it is desirable to arrange the order and combination of the processes with a view to obtain them as much as possible *symmetrically* and in cycles, in order that the mechanical advantages of the *backing* system may be applied to the utmost. It is here interesting to observe the manner in which the value of an *analytical* resource is *met* and *enhanced* by an ingenious *mechanical* contrivance. We see in it an instance of one of those mutual *adjustments* between the purely mathematical and the mechanical departments, mentioned in Note A as being a main and essential condition of success in the invention of a calculating engine. The nature of the resources afforded by such adjustments would be of two principal kinds. In some cases, a difficulty (perhaps in itself insurmountable) in the one department, would be overcome by facilities in the other; and sometimes (as in the present case) a strong point in the one, would be rendered still stronger and more available, by combination with a corresponding strong point in the other.

As a mere example of the degree to which the combined systems of cycles and of backing can diminish the *number* of cards requisite, we shall choose a case which places it in strong evidence, and which has likewise the advantage of being a perfectly different *kind* of problem from those that are mentioned in any of the other Notes. Suppose it be required to eliminate nine variables from ten simple equations of the form—

$$ax_0 + bx_1 + cx_2 + dx_3 + \dots = p \qquad (1)$$

$$a^1x_0 + b^1x_1 + c^1x_2 + d^1x_3 + \dots = p^1 \qquad (2)$$

etc. etc. etc.

We should explain, before proceeding, that it is not our object to consider this problem with reference to the actual arrangement of the data on the Variables of the engine, but simply as an abstract question of the *nature* and *number* of the *operations* required to be performed during its complete solution.

The first step would be the elimination of the first unknown quantity x_0 between the two first equations. This would be obtained by the form—

$$(a^1a - aa^1)x_0 + (a^1b - ab^1)x_1 + (a^1c - ac^1)x_2 + (a^1d - ad^1)x_3 + \dots = a^1p - ap^1$$

for which the operations $10(\times, \times, -)$ would be needed. The second step would be the elimination of x_0 between the second and third equations, for which the operations would be precisely the same. We should then have had altogether the following operations—

$$10(\times, \times, -),\ 10(\times, \times, -) = 20(\times, \times, -)$$

Continuing in the same manner, the total number of operations for the complete elimination of x_0 between all the successive pairs of equations, would be—

$$9 \cdot 10(\times, \times, -) = 90(\times, \times, -)$$

We should then be left with nine simple equations of nine variables from which to eliminate the next variable x_1; for which the total of the processes would be—

$$8 \cdot 9(\times, \times, -) = 72(\times, \times, -)$$

We should then be left with eight simple equations of eight variables from which to eliminate x_2, for which the processes would be—

$$7 . 8(\times, \times, -) = 56(\times, \times, -)$$

and so on. The total operations for the elimination of all the variables would thus be—

$$9 . 10 + 8 . 9 + 7 . 8 + 6 . 7 + 5 . 6 + 4 . 5 + 3 . 4 + 2 . 3 + 1 . 2 = 330$$

So that *three* Operation-cards would perform the office of 330 such cards.

If we take n simple equations containing $n - 1$ variables, n being a number unlimited in magnitude, the case becomes still more obvious, as the same three cards might then take the place of thousands or millions of cards.

We shall now draw further attention to the fact, already noticed, of its being by no means necessary that a formula proposed for solution should ever have been actually worked out, as a condition for enabling the engine to solve it. Provided we know the *series of operations* to be gone through, that is sufficient. In the foregoing instance this will be obvious enough on a slight consideration. And it is a circumstance which deserves particular notice, since herein may reside a latent value of such an engine almost incalculable in its possible ultimate results. We already know that there are functions whose numerical value it is of importance for the purposes both of abstract and of practical science to ascertain, but whose determination requires processes so lengthy and so complicated, that, although it is possible to arrive at them through great expenditure of time, labour and money, it is yet on these accounts practically almost unattainable; and we can conceive there being some results which it may be *absolutely impossible* in practice to attain with any accuracy, and whose precise determination it may prove highly important for some of the future wants of science in its manifold, complicated and rapidly-developing fields of inquiry, to arrive at.

Without, however, stepping into the region of conjecture, we will mention a particular problem which occurs to us at this moment as being an apt illustration of the use to which such an engine may be turned for determining that which human brains find it difficult or impossible to work out unerringly. In the solution of the famous problem of the Three Bodies, there are, out of about 295 coefficients of lunar perturbations given by M. Clausen (Astro^{e}. Nachrichten, No. 406) as the result of the calculations by Burg, of two by Damoiseau, and of one by Burckhardt, fourteen coefficients that differ in the nature of their algebraic sign; and out of the remainder there are only 101 (or about one-third) that agree precisely both in signs and in amount. These discordances, which are generally small in individual magnitude, may arise either from an erroneous determination of the abstract coefficients in the development of the problem, or from discrepancies in the data deduced from observation, or from both causes combined. The former is the most ordinary source of error in astronomical computations, and this the engine would entirely obviate.

We might even invent laws for series or formulae in an arbitrary manner, and set the engine to work upon them, and thus deduce numerical results which we might not otherwise have thought of obtaining. But this would hardly perhaps in any instance be productive of any great practical utility, or calculated to rank higher than as a kind of philosophical amusement. A. A. L.

Note G—Page 360

It is desirable to guard against the possibility of exaggerated ideas that might arise as to the powers of the Analytical Engine. In considering any new subject, there is frequently a tendency, first, to *overrate* what we find to be already interesting or remarkable; and, secondly, by a sort of natural reaction, to *undervalue* the true state of the case, when we do discover that our notions have surpassed those that were really tenable.

The Analytical Engine has no pretensions whatever to *originate* anything. It can do whatever we *know how to order it* to perform. It can *follow* analysis; but it has no power of *anticipating* any analytical relations or truths. Its province is to assist us in making *available* what we are already acquainted with. This it is calculated to effect primarily and chiefly of course, through its executive faculties; but it is likely to exert an *indirect* and reciprocal influence on science itself in another manner. For, in so distributing and combining the truths and the formulae of analysis, that they may become most easily and rapidly amenable to the mechanical combinations of the engine, the relations and the nature of many subjects in that science are necessarily thrown into new lights, and more profoundly investigated. This is a decidedly indirect, and a somewhat *speculative*, consequence of such an invention. It is, however, pretty evident, on general principles, that in devising for mathematical truths a new form in which to record and throw themselves out for actual use, views are likely to be induced, which should again react on the more theoretical phase of the subject. There are in all extensions of human power, or additions to human knowledge, various *collateral* influences, besides the main and primary object attained.

To return to the executive faculties of this engine: the question must arise in every mind, are they *really* even able to *follow* analysis in its whole extent? No reply, entirely satisfactory to all minds, can be given to this query, excepting the actual existence of the engine, and actual experience of its practical results. We will, however, sum up for each reader's consideration the chief elements with which the engine works—

1. It performs the four operations of simple arithmetic upon any numbers whatever.

2. By means of certain artifices and arrangements (upon which we cannot enter within the restricted space which such a publication as the present may admit of), there is no limit either to the *magnitude* of the *numbers* used, or to the *number* of *quantities* (either variables or constants) that may be employed.

3. It can combine these numbers and these quantities either algebraically or arithmetically, in relations unlimited as to variety, extent, or complexity.

4. It uses algebraic *signs* according to their proper laws, and develops the logical consequences of these laws.

5. It can arbitrarily substitute any formula for any other; effacing the first from the columns on which it is represented, and making the second appear in its stead.

6. It can provide for singular values. Its power of doing this is referred to in M. Menabrea's memoir, page 359, where he mentions the passage of values through zero and infinity. The practicability of causing it arbitrarily to change its processes at any moment, on the occurrence of any specified contingency (of which its substitution of ($\frac{1}{2}\cos . \overline{n+1}\theta + \frac{1}{2}\cos . \overline{n-1}\theta$) for ($\cos n\theta . \cos \theta$) explained in Note E, is in some degree an illustration), at once secures this point.

The subject of integration and of differentiation demands some notice. The engine can effect these processes in either of two ways—

First, we may order it, by means of the Operation and of the Variable-cards, to go through the various steps by which the required *limit* can be worked out for whatever function is under consideration.

Secondly, it may (if we know the form of the limit for the function in question) effect the integration or differentiation by direct* substitution. We remarked in Note B, that any *set* of columns on which numbers are inscribed, represents merely a *general* function of the several quantities, until the special function have been impressed by means of the Operation and Variable-cards. Consequently, if instead of requiring the value of the function, we require that of its integral, or of its differential coefficient, we have merely to order whatever particular combination of the ingredient

* The engine cannot of course compute limits for perfectly *simple* and *uncompounded* functions, except in this manner. It is obvious that it has no power of representing or of manipulating with any but *finite* increments or decrements; and consequently that wherever the computation of limits (or of any other functions) depends upon the *direct* introduction of quantities which either increase or decrease *indefinitely*, we are absolutely beyond the sphere of its powers. Its nature and arrangements are remarkably adapted for taking into account all *finite* increments or decrements (however small or large), and for developing the true and logical modifications of form or value dependent upon differences of this nature. The engine may indeed be considered as including the whole Calculus of Finite Differences; many of whose theorems would be especially and beautifully fitted for development by its processes, and would offer peculiarly interesting considerations. We may mention, as an example, the calculation of the Numbers of Bernoulli by means of the *Differences of Nothing*.

quantities may constitute that integral or that coefficient. In ax^n, for instance, instead of the quantities

V_0	V_1	V_2	V_3
a	n	x	ax^n

$$\underbrace{a \quad n \quad x}_{ax^n}$$

being ordered to appear on V_3 in the combination ax^n, they would be ordered to appear in that of

$$anx^{n-1}$$

They would then stand thus—

V_0	V_1	V_2	V_3
a	n	x	anx^{n-1}

$$\underbrace{a \quad n \quad x}_{anx^{n-1}}$$

Similarly, we might have $\frac{a}{n}x^{n+1}$, the integral of ax^n.

An interesting example for following out the processes of the engine would be such a form as—

$$\int \frac{x^n dx}{\sqrt{a^2 - x^2}}$$

or any other cases of integration by successive reductions, where an integral which contains an operation repeated n times can be made to depend upon another which contains the same $n - 1$ or $n - 2$ times, and so on until by continued reduction we arrive at a certain *ultimate* form, whose value has then to be determined.

The methods in Arbogast's *Calcul des Dérivations* are peculiarly fitted for the notation and the processes of the engine. Likewise the whole of the Combinatorial Analysis, which consists first in a purely numerical calculation of indices, and secondly in the distribution and combination of the quantities according to laws prescribed by these indices.

We will terminate these Notes by following up in detail the steps through which the engine could compute the Numbers of Bernoulli, this being (in the form in which we shall deduce it) a rather complicated example of its powers. The simplest manner of computing these numbers would be from the direct expansion of

$$\frac{x}{\varepsilon^x - 1} = \frac{1}{1 + \frac{x}{2} + \frac{x^2}{2 \cdot 3} + \frac{x^3}{2 \cdot 3 \cdot 4} + \text{etc.}} \quad . \quad . \quad (1)$$

which is in fact a particular case of the development of

$$\frac{a + bx + cx^2 + \text{etc.}}{a' + b'x + c'x^2 + \text{etc.}}$$

mentioned in Note E. Or again, we might compute them from the well-known form

$$B_{2n-1} = 2 \cdot \frac{1 \cdot 2 \cdot 3 \cdot \ldots \cdot 2n}{(2\pi)^{2n}} \cdot \left\{1 + \frac{1}{2^{2n}} + \frac{1}{3^{2n}} + \ldots\right\} \quad . \quad (2)$$

or from the form

$$B_{2n-1} = \frac{\pm 2n}{(2^{2n} - 1)2^{n-1}} \left\{ \begin{array}{l} \frac{1}{2} n^{2n-1} \\ -(n-1)^{2n-1}\left\{1 + \frac{1}{2} \cdot \frac{2n}{1}\right\} \\ +(n-2)^{2n-1}\left\{1 + \frac{2n}{1} + \frac{1}{2} \cdot \frac{2n \cdot (2n-1)}{1 \cdot 2}\right\} \\ -(n-3)^{2n-1}\left\{\begin{array}{l} 1 + \frac{2n}{1} + \frac{2n \cdot (2n-1)}{1 \cdot 2} \\ \quad + \frac{1}{2} \cdot \frac{2n \cdot (2n-1) \cdot (2n-2)}{1 \cdot 2 \cdot 3} \end{array}\right\} \\ + \ldots \quad \ldots \quad \ldots \quad \ldots \end{array} \right\} \quad (3)$$

or from many others. As, however, our object is not simplicity or facility of computation, but the illustration of the powers of the engine, we prefer selecting the formula below, marked (8). This is derived in the following manner.

If in the equation

$$\frac{x}{\varepsilon^x - 1} = 1 - \frac{x}{2} + B_1\frac{x^2}{2} + B_3\frac{x^4}{2 \cdot 3 \cdot 4} + B_5\frac{x^6}{2 \cdot 3 \cdot 4 \cdot 5 \cdot 6} + \quad . \quad (4)$$

(in which B_1, B_3, . . ., etc., are the Numbers of Bernoulli), we expand the denominator of the first side in powers of x, and then divide both numerator and denominator by x, we shall derive

$$1 = \left(1 - \frac{x}{2} + B_1\frac{x^2}{2} + B_3\frac{x^4}{2 \cdot 3 \cdot 4} + \ldots\right)\left(1 + \frac{x}{2} + \frac{x^2}{2 \cdot 3} + \frac{x^3}{2 \cdot 3 \cdot 4} \ldots\right) (5)$$

If this latter multiplication be actually performed, we shall have a series of the general form

$$1 + D_1x + D_2x^2 + D_3x^3 + \ldots \quad . \quad . \quad (6)$$

in which we see, first, that all the coefficients of the powers of x are severally equal to zero; and secondly, that the general form for D_{2n}, the coefficient of the $2n+1$th *term* (that is of x^{2n} any *even* power of x), is the following—

$$\left.\begin{aligned}\frac{1}{2\,.\,3\,...\,2n+1}-\frac{1}{2}\,.\,\frac{1}{2\,.\,3\,...\,2n}+\frac{B_1}{2}\,.\,\frac{1}{2\,.\,3\,...\,2n-1}+\\ \frac{B_3}{2\,.\,3\,.\,4}\,.\,\frac{1}{2\,.\,3\,...\,2n-3}+\frac{B_5}{2\,.\,3\,.\,4\,.\,5\,.\,6}\,.\,\frac{1}{2\,.\,3\,...\,2n-5}\\ ...+\frac{B_{2n-1}}{2\,.\,3\,...\,2n}\,.\,1=0\end{aligned}\right\}\quad(7)$$

Multiplying every term by $(2\,.\,3\,...\,2n)$, we have

$$\left.\begin{aligned}0=-\frac{1}{2}\,.\,\frac{2n-1}{2n+1}+B_1\left(\frac{2n}{2}\right)+B_3\left(\frac{2n\,.\,2n-1\,.\,2n-2}{2\,.\,3\,.\,4}\right)\\ +B_5\left(\frac{2n\,.\,2n-1\,...\,2n-4}{2\,.\,3\,.\,4\,.\,5\,.\,6}\right)+...+B_{2n-1}\end{aligned}\right\}\quad(8)$$

which it may be convenient to write under the general form—

$$0=A_0+A_1B_1+A_3B_3+A_5B_5+...+B_{2n-1}\,...\quad(9)$$

A_1, A_3, etc., being those functions of n which respectively belong to B_1, B_3, etc.

We might have derived a form nearly similar to (8), from D_{2n-1} the coefficient of any *odd* power of x in (6); but the general form is a little different for the coefficients of the *odd* powers, and not quite so convenient.

On examining (7) and (8), we perceive that, when these formulae are isolated from (6) whence they are derived, and considered in themselves separately and independently, n may be any whole number whatever; although when (7) occurs *as one of the D*'s in (6), it is obvious that n is then not arbitrary, but is always a certain function of the *distance of that D from the beginning*. If that distance be $= d$, then

$$2n+1=d,\text{ and } n=\frac{d-1}{2}\text{ (for any *even* power of } x)$$

$$2n=d,\text{ and } n=\frac{d}{2}\text{ (for any *odd* power of } x)$$

It is with the *independent* formula (8) that we have to do. There ore it must be remembered that the conditions for the value of n are now modified, and that n is a perfectly *arbitrary* whole number. This circumstance, combined with the fact (which we may easily perceive) that whatever n is, every term of (8) after the $(n+1)$th is $=0$, and that the $(n+1)$th term itself is always $=B_{2n-1}\,.\,\frac{1}{1}=B_{2n-1}$, enables us to find the value (either numerical or algebraical) of any nth Number of Bernoulli B_{2n-1}, *in terms of all the* *preceding ones*, if we but know the values of B_1, B_3, . . . B_{2n-3}. We append to this Note a Diagram and Table, containing the details of the computation for B_7, (B_1, B_3, B_5 being supposed given).

On attentively considering (8), we shall likewise perceive that we may derive from it the numerical value of *every* Number of Bernoulli in succession, from the very beginning, *ad infinitum*, by the following series of computations—

1st Series.—Let $n = 1$, and calculate (8) for this value of n. The result is B_1.

2nd Series.—Let $n = 2$. Calculate (8) for this value of n, substituting the value of B_1 just obtained. The result is B_3.

3rd Series.—Let $n = 3$. Calculate (8) for this value of n, substituting the values of B_1, B_3 before obtained. The result is B_5. And so on, to any extent.

The diagram* represents the columns of the engine when just prepared for computing B_{2n-1} (in the case of $n = 4$); while the table beneath them presents a complete simultaneous view of all the successive changes which these columns then severally pass through in order to perform the computation. (The reader is referred to Note D, for explanations respecting the nature and notation of such tables.)

Six numerical *data* are in this case necessary for making the requisite combinations. These data are 1, 2, $n(= 4)$, B_1, B_3, B_5. Were $n = 5$, the additional datum B_7 would be needed. Were $n = 6$, the datum B_9 would be needed; and so on. Thus the actual *number of data* needed will always be $n + 2$, for $n = n$; and out of these $n + 2$ data, $(\overline{n + 2} - 3)$ of them are successive Numbers of Bernoulli. The reason why the Bernoulli Numbers used as data, are nevertheless placed on *Result*-columns in the diagram, is because they may properly be supposed to have been previously computed in succession by the *engine* itself; under which circumstances each B will appear as a *result*, previous to being used as a *datum* for computing the succeeding B. Here then is an instance (of the kind alluded to in Note D) of the same Variables filling more than one office in turn. It is true that if we consider our computation of B_7 as a perfectly *isolated* calculation, we may conclude B_1, B_3, B_5 to have been arbitrarily placed on the columns; and it would then perhaps be more consistent to put them on V_4, V_5, V_6 as data and not results. But we are not taking this view. On the contrary, we suppose the engine to be *in the course of* computing the Numbers to an indefinite extent, from the very beginning; and that we merely single out, by way of example, *one amongst* the successive but distinct series of computations it is thus performing. Where the B's are fractional, it must be understood that they are computed and appear in the notation of *decimal* fractions. Indeed this is a circumstance that should be noticed with reference to all calculations. In any of the examples

* See the diagram facing page 404.

already given in the translation and in the Notes, some of the *data*, or of the temporary or permanent results, might be fractional, quite as probably as whole numbers. But the arrangements are so made, that the nature of the processes would be the same as for whole numbers.

In the above table and diagram we are not considering the *signs* of any of the *B*'s, merely their numerical magnitude. The engine would bring out the sign for each of them correctly of course, but we cannot enter on *every* additional detail of this kind, as we might wish to do. The circles for the signs are therefore intentionally left blank in the diagram.

Operation-cards 1, 2, 3, 4, 5, 6 prepare $-\frac{1}{2}\cdot\frac{2n-1}{2n+1}$. Thus, Card 1 multiplies *two* into n, and the three *Receiving* Variable-cards belonging respectively to V_4, V_5, V_6, allow the result $2n$ to be placed on each of these latter columns (this being a case in which a triple receipt of the result is needed for subsequent purposes); we see that the upper indices of the two Variables used, during Operation 1, remain unaltered.

We shall not go through the details of every operation singly, since the table and diagram sufficiently indicate them; we shall merely notice some few peculiar cases.

By Operation 6, a *positive* quantity is turned into a *negative* quantity, by simply subtracting the quantity from a column which has only zero upon it. (The sign at the top of V_{13} would become $-$ during this process.)

Operation 7 will be unintelligible, unless it be remembered that if we were calculating for $n = 1$ instead of $n = 4$, Operation 6 would have completed the computation of B_1 itself; in which case the engine, instead of continuing its processes, would have to put B_1 on V_{21}; and then either to stop altogether, or to begin Operations 1, 2, . . . 7 all over again for value of $n(= 2)$, in order to enter on the computation of B_3; (having, however, taken care, previous to this recommencement, to make the number on V_3 equal to *two*, by the addition of unity to the former $n = 1$ on that column). Now Operation 7 must either bring out a result equal to zero (if $n = 1$); or a result *greater* than *zero*, as in the present case; and the engine follows the one or the other of the two courses just explained, contingently on the one or the other result of Operation 7. In order fully to perceive the necessity of this *experimental* operation, it is important to keep in mind what was pointed out, that we are not treating a perfectly isolated and independent computation, but one out of a series of antecedent and prospective computations.

Cards 8, 9, 10 produce $-\frac{1}{2}\cdot\frac{2n-1}{2n+1} + B_1\frac{2n}{2}$. In Operation 9 we see an example of an upper index which again becomes a value after having passed from preceding values to zero. V_{11} has successively been ${}^0V_{11}$, ${}^1V_{11}$, ${}^2V_{11}$, ${}^0V_{11}$, ${}^3V_{11}$; and, from the nature of the office which V_{11} performs in the calculation, its index will continue to go through further changes of the

DIAGRAM FOR THE COMPUTATION BY THE ENGINE OF THE NUMBERS OF BERNOULLI. (SEE NOTE G.)

Number of Operation	Nature of Operation	Variables Acted Upon	Variables Receiving Results	Indication of change in the Value on any Variable	Statement of Results	Data: $^1V_1$	Data: $^1V_2$	Data: $^1V_3$	Working Variables: $^0V_4$	$^0V_5$	$^0V_6$	$^0V_7$	$^0V_8$	$^0V_9$	$^0V_{10}$	$^0V_{11}$	$^0V_{12}$	$^0V_{13}$	Result Variables: $^1V_{21}$ B_1 in a decimal fraction	$^1V_{22}$ B_3 in a decimal fraction	$^1V_{23}$ B_5 in a decimal fraction	$^0V_{24}$...
						○ 0 0 0 1	○ 0 0 0 2	○ 0 0 0 4	○ 0 0 0 0	○ 0 0 0 0	○ 0 0 0 0	○ 0 0 0 0	○ 0 0 0 0	○ 0 0 0 0	○ 0 0 0 0	○ 0 0 0 0	○ 0 0 0 0	○ 0 0 0 0	○	○	○	○ 0 0 0
						1	2	n											B_1	B_3	B_5	B_7
1	×	$^1V_2 \times {}^1V_3$	$^1V_4, {}^1V_5, {}^1V_6$	$\begin{Bmatrix}{}^1V_2 = {}^1V_2 \\ {}^1V_3 = {}^1V_3\end{Bmatrix}$	$= 2n$	—	2	n	$2n$	$2n$	$2n$											
2	−	$^1V_4 - {}^1V_1$	$^2V_4$	$\begin{Bmatrix}{}^1V_4 = {}^2V_4 \\ {}^1V_1 = {}^1V_1\end{Bmatrix}$	$= 2n - 1$	1	—	—	$2n - 1$													
3	+	$^1V_5 + {}^1V_1$	$^2V_5$	$\begin{Bmatrix}{}^1V_5 = {}^2V_5 \\ {}^1V_1 = {}^1V_1\end{Bmatrix}$	$= 2n + 1$	1	—	—	—	$2n + 1$												
4	÷	$^2V_5 \div {}^2V_4$	$^1V_{11}$	$\begin{Bmatrix}{}^2V_5 = {}^0V_5 \\ {}^2V_4 = {}^0V_4\end{Bmatrix}$	$= \frac{2n-1}{2n+1}$	—	—	—	0	0	—	—	—	—	—	$\frac{2n-1}{2n+1}$						
5	÷	$^1V_{11} \div {}^1V_2$	$^2V_{11}$	$\begin{Bmatrix}{}^1V_{11} = {}^2V_{11} \\ {}^1V_2 = {}^1V_2\end{Bmatrix}$	$= \frac{1}{2} \cdot \frac{2n-1}{2n+1}$	—	2	—	—	—	—	—	—	—	—	$\frac{1}{2} \cdot \frac{2n-1}{2n+1}$						
6	−	$^0V_{13} - {}^2V_{11}$	$^1V_{13}$	$\begin{Bmatrix}{}^2V_{11} = {}^0V_{11} \\ {}^0V_{13} = {}^1V_{13}\end{Bmatrix}$	$= -\frac{1}{2} \cdot \frac{2n-1}{2n+1} = A_0$	—	—	—	—	—	—	—	—	—	—	0	—	$-\frac{1}{2} \cdot \frac{2n-1}{2n+1} = A_0$				
7	−	$^1V_3 - {}^1V_1$	$^1V_{10}$	$\begin{Bmatrix}{}^1V_3 = {}^1V_3 \\ {}^1V_1 = {}^1V_1\end{Bmatrix}$	$= n - 1 (= 3)$	1	—	n	—	—	—	—	—	—	$n - 1$							
8	+	$^1V_2 + {}^0V_7$	$^1V_7$	$\begin{Bmatrix}{}^1V_2 = {}^1V_2 \\ {}^0V_7 = {}^1V_7\end{Bmatrix}$	$= 2 + 0 = 2$	—	2	—	—	—	—	2										
9	÷	$^1V_6 \div {}^1V_7$	$^3V_{11}$	$\begin{Bmatrix}{}^1V_6 = {}^1V_6 \\ {}^0V_{11} = {}^3V_{11}\end{Bmatrix}$	$= \frac{2n}{2} = A_1$	—	—	—	—	—	$2n$	2	—	—	—	$\frac{2n}{2} = A_1$						
10	×	$^1V_{21} \times {}^3V_{11}$	$^1V_2$	$\begin{Bmatrix}{}^1V_{21} = {}^1V_{21} \\ {}^3V_{11} = {}^3V_{11}\end{Bmatrix}$	$= B_1 \cdot \frac{2n}{2} = B_1 A_1$	—	—	—	—	—	—	—	—	—	—	$\frac{2n}{2} = A_1$	$B_1 \cdot \frac{2n}{2} = B_1 A_1$	—	B_1			
11	+	$^1V_{12} + {}^1V_{13}$	$^2V_{13}$	$\begin{Bmatrix}{}^1V_{12} = {}^0V_{12} \\ {}^1V_{13} = {}^2V_{13}\end{Bmatrix}$	$= -\frac{1}{2} \cdot \frac{2n-1}{2n+1} + B_1 \cdot \frac{2n}{2}$	—	—	—	—	—	—	—	—	—	—	—	0	$\left(-\frac{1}{2} \cdot \frac{2n-1}{2n+1} + B_1 \cdot \frac{2n}{2}\right)$				
12	−	$^1V_{10} - {}^1V_1$	$^2V_{10}$	$\begin{Bmatrix}{}^1V_{10} = {}^2V_{10} \\ {}^1V_1 = {}^1V_1\end{Bmatrix}$	$= n - 2 (= 2)$	1	—	—	—	—	—	—	—	—	$n - 2$							
13	−	$^1V_6 - {}^1V_1$	$^2V_6$	$\begin{Bmatrix}{}^1V_6 = {}^2V_6 \\ {}^1V_1 = {}^1V_1\end{Bmatrix}$	$= 2n - 1$	1	—	—	—	—	$2n - 1$											
14	+	$^1V_1 + {}^1V_7$	$^2V_7$	$\begin{Bmatrix}{}^1V_1 = {}^1V_1 \\ {}^1V_7 = {}^2V_7\end{Bmatrix}$	$= 2 + 1 = 3$	1	—	—	—	—	—	3										
15	÷	$^2V_6 \div {}^2V_7$	$^1V_8$	$\begin{Bmatrix}{}^2V_6 = {}^2V_6 \\ {}^2V_7 = {}^2V_7\end{Bmatrix}$	$= \frac{2n-1}{3}$	—	—	—	—	—	$2n - 1$	3	$\frac{2n-1}{3}$									
16	×	$^1V_8 \times {}^3V_{11}$	$^4V_{11}$	$\begin{Bmatrix}{}^1V_8 = {}^0V_8 \\ {}^3V_{11} = {}^4V_{11}\end{Bmatrix}$	$= \frac{2n}{2} \cdot \frac{2n-1}{3}$	—	—	—	—	—	—	—	0	—	—	$\frac{2n}{2} \cdot \frac{2n-1}{3}$						
17	−	$^2V_6 - {}^1V_1$	$^3V_6$	$\begin{Bmatrix}{}^2V_6 = {}^3V_6 \\ {}^1V_1 = {}^1V_1\end{Bmatrix}$	$= 2n - 2$	1	—	—	—	—	$2n - 2$											
18	+	$^1V_1 + {}^2V_7$	$^3V_7$	$\begin{Bmatrix}{}^2V_7 = {}^3V_7 \\ {}^1V_1 = {}^1V_1\end{Bmatrix}$	$= 3 + 1 = 4$	1	—	—	—	—	—	4										
19	÷	$^3V_6 \div {}^3V_7$	$^1V_9$	$\begin{Bmatrix}{}^3V_6 = {}^3V_6 \\ {}^3V_7 = {}^3V_7\end{Bmatrix}$	$= \frac{2n-2}{4}$	—	—	—	—	—	$2n - 2$	4	—	$\frac{2n-2}{4}$	—							
20	×	$^1V_9 \times {}^4V_{11}$	$^5V_{11}$	$\begin{Bmatrix}{}^1V_9 = {}^0V_9 \\ {}^4V_{11} = {}^5V_{11}\end{Bmatrix}$	$= \frac{2n}{2} \cdot \frac{2n-1}{3} \cdot \frac{2n-2}{4} = A_3$	—	—	—	—	—	—	—	—	0		$\left\{\frac{2n}{2} \cdot \frac{2n-1}{3} \cdot \frac{2n-2}{4} = A_3\right\}$						
21	×	$^1V_{22} \times {}^5V_{11}$	$^2V_{12}$	$\begin{Bmatrix}{}^1V_{22} = {}^1V_{22} \\ {}^0V_{12} = {}^2V_{12}\end{Bmatrix}$	$= B_3 \cdot \frac{2n}{2} \cdot \frac{2n-1}{3} \cdot \frac{2n-2}{4} = B_3 A_3$	—	—	—	—	—	—	—	—	—	—	0	$B_3 A_3$	—	—	B_3		
22	+	$^2V_{12} + {}^2V_{13}$	$^3V_{13}$	$\begin{Bmatrix}{}^2V_{12} = {}^0V_{12} \\ {}^2V_{13} = {}^3V_{13}\end{Bmatrix}$	$= A_0 + B_1 A_1 + B_3 A_3$	—	—	—	—	—	—	—	—	—	—	—	0	$\{A_0 + B_1 A_1 + B_3 A_3\}$				
23	−	$^2V_{10} - {}^1V_1$	$^3V_{10}$	$\begin{Bmatrix}{}^2V_{10} = {}^3V_{10} \\ {}^1V_1 = {}^1V_1\end{Bmatrix}$	$= n - 3 (= 1)$	1	—	—	—	—	—	—	—	—	$n - 3$							
					Here follows a repetition of Operations thirteen to twenty-three																	
24	+	$^4V_{13} + {}^0V_{24}$	$^1V_{24}$	$\begin{Bmatrix}{}^4V_{13} = {}^0V_{13} \\ {}^0V_{24} = {}^1V_{24}\end{Bmatrix}$	$= B_7$	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	B_7
25	+	$^1V_1 + {}^1V_3$	$^1V_3$	$\begin{Bmatrix}{}^1V_1 = {}^1V_1 \\ {}^1V_3 = {}^1V_3 \\ {}^5V_6 = {}^0V_6 \\ {}^5V_7 = {}^0V_7\end{Bmatrix}$	$= n + 1 = 4 + 1 = 5$ by a Variable-card by a Variable-card	1	—	$n + 1$	—	—	0	0										

same description, which, if examined, will be found to be regular and periodic.

Card 12 has to perform the same office as Card 7 did in the preceding section; since, if n had been $= 2$, the 11th operation would have completed the computation of B_3.

Cards 13 to 20 make A_3. Since A_{2n-1} always consists of $2n - 1$ factors, A_2 has three factors; and it will be seen that Cards 13, 14, 15, 16 make the second of these factors, and then multiply it with the first; and that 17, 18, 19, 20 make the third factor, and then multiply this with the product of the two former factors.

Card 23 has the office of Cards 11 and 7 to perform, since if n were $= 3$, the 21st and 22nd operations would complete the computation of B_5. As our case is B_7, the computation will continue one more stage; and we must now direct attention to the fact, that in order to compute A_7 it is merely necessary precisely to repeat the group of Operations 13 to 20; and then, in order to complete the computation of B_7, to repeat Operations 21, 22.

It will be perceived that every unit added to n in B_{2n-1}, entails an additional repetition of operations (13, . . . 23) for the computation of B_{2n-1}. Not only are all the *operations* precisely the same, however, for every such repetition, but they require to be respectively supplied with numbers from the very *same pairs of columns*; with only the one exception of Operation 21, which will of course need B_5 (from V_{23}) instead of B_3 (from V_{22}). This identity in the *columns* which supply the requisite numbers, must not be confounded with identity in the *values* those columns have upon them and give out to the mill. Most of those values undergo alterations during a performance of the operations (13, . . . 23), and consequently the columns present a new set of values for the *next* performance of (13, . . . 23) to work on.

At the termination of the *repetition* of operations (13, . . . 23) in computing B_7, the alterations in the values on the Variables are, that

$V_6 = 2n - 4$ instead of $2n - 2$.

$V_7 = 6$ . . . . 4.

$V_{10} = 0$ . . . . 1.

$V_{13} = A_0 + A_1B_1 + A_3B_3 + A_5B_5$ instead of $A_0 + A_1B_1 + A_3B_3$

In this state the only remaining processes are first: to transfer the value which is on V_{13}, to V_{24}; and secondly to reduce V_6, V_7, V_{13} to zero, and to add* *one* to V_3, in order that the engine may be ready to commence computing B_9. Operations 24 and 25 accomplish these purposes. It may be

* It is interesting to observe, that so complicated a case as this calculation of the Bernoullian Numbers, nevertheless, presents a remarkable simplicity in one respect; viz. that during the processes for the computation of *millions* of these Numbers, no other arbitrary modification would be requisite in the arrangements, excepting the above simple and uniform provision for causing one of the data periodically to receive the finite increment unity.

thought anomalous that Operation 25 is represented as leaving the upper index of V_3 still = unity. But it must be remembered that these indices always begin anew for a separate calculation, and that Operation 25 places upon V_3 the *first* value *for the new calculation.*

It should be remarked, that when the group (13, . . . 23) is *repeated*, changes occur in some of the *upper* indices during the course of the repetition: for example, ${}^3V_6$ would become ${}^4V_6$ and ${}^5V_6$.

We thus see that when $n = 1$, nine Operation-cards are used; that when $n = 2$, fourteen Operation-cards are used; and that when $n > 2$, twenty-five Operation-cards are used; but that no *more* are needed, however great n may be; and not only this, but that these same twenty-five cards suffice for the successive computation of all the Numbers from B_1 to B_{2n-1} inclusive. With respect to the number of *Variable*-cards, it will be remembered, from the explanations in previous Notes, that an average of three such cards to each *operation* (not, however, to each Operation-*card*) is the estimate. According to this the computation of B_1 will require twenty-seven Variable-cards; B_3 forty-two such cards; B_5 seventy-five; and for every succeeding B after B_5, there would be thirty-three additional Variable-cards (since each repetition of the group (13, . . . 23) adds eleven to the number of operations required for computing the previous B). But we must now explain, that whenever there is a *cycle of operations*, and if these merely require to be supplied with numbers from the *same pairs of columns* and likewise each operation to place its *result* on the *same* column for every repetition of the whole group, the process then admits of a *cycle of Variable-cards* for effecting its purposes. There is obviously much more symmetry and simplicity in the arrangements, when cases do admit of repeating the Variable as well as the Operation-cards. Our present example is of this nature. The only exception to a *perfect identity* in *all* the processes and columns used, for every repetition of Operations (13, . . . 23) is, that Operation 21 always requires one of its factors from a new column, and Operation 24 always puts its result on a new column. But as these variations follow the same law at each repetition (Operation 21 always requiring its factor from a column *one* in advance of that which it used the previous time, and Operation 24 always putting its result on the column *one* in advance of that which received the previous result), they are easily provided for in arranging the recurring group (or cycle) of Variable-cards.

We may here remark that the average estimate of three Variable-cards coming into use to each operation, is not to be taken as an absolutely and literally correct amount for all cases and circumstances. Many special circumstances, either in the nature of a problem, or in the arrangements of the engine under certain contingencies, influence and modify this average to a greater or less extent. But it is a very safe and correct *general* rule to go upon. In the preceding case it will give us seventy-five Variable-cards as the total number which will be necessary for computing any B after B_3.

This is very nearly the precise amount really used, but we cannot here enter into the minutiae of the few particular circumstances which occur in this example (as indeed at some one stage or other of probably most computations) to modify slightly this number.

It will be obvious that the very *same* seventy-five Variable-cards may be repeated for the computation of every succeeding Number, just on the same principle as admits of the repetition of the thirty-three Variable-cards of Operations (13, . . . 23) in the computation of any *one* Number. Thus there will be a *cycle of a cycle* of Variable-cards.

If we now apply the notation for cycles, as explained in Note E, we may express the operations for computing the Numbers of Bernoulli in the following manner—

$(1 \ldots 7), (24, 25)$	gives B_1	= 1st	number;	(n being	= 1)
$(1 \ldots 7), (8 \ldots 12), (24, 25)$	„ B_3	= 2nd	„ ;	(n „	= 2)
$(1 \ldots 7), (8 \ldots 12), (13 \ldots 23), (24, 25)$	„ B_5	= 3rd	„ ;	(n „	= 3)
$(1 \ldots 7), (8 \ldots 12), 2(13 \ldots 23), (24, 25)$	„ B_7	= 4th	„ ;	(n „	= 4)

. . . .

. . . .

$(1 \ldots 7), (8 \ldots 12), \Sigma(+1)^{n-2}(13 \ldots 23), (24, 25) \ldots B_{2n-1}$

= nth „ ; (n „ = n)

Again,

$$(1 \ldots 7), (24, 25), \underset{\text{limits 1 to } n}{\Sigma(+1)^n}\{(1 \ldots 7), (8 \ldots 12), \underset{\text{limits 0 to } (n+2)}{\Sigma(n+2)} (13 \ldots 23), (24, 25)\}$$

represents the total operations for computing every number in succession, from B_1 to B_{2n-1} inclusive.

In this formula we see a *varying cycle* of the *first* order, and an ordinary cycle of the *second* order. The latter cycle in this case includes in it the varying cycle.

On inspecting the ten Working-Variables of the diagram, it will be perceived, that although the *value* on any one of them (excepting V_4 and V_5) goes through a series of changes, the *office* which each performs is in this calculation *fixed* and *invariable*. Thus V_6 always prepares the *numerators* of the factors of any A; V_7 the *denominators*. V_8 always receives the $(2n-3)$th factor of A_{2n-1}, and V_9 the $(2n-1)$th. V_{10} always decides which of two courses the succeeding processes are to follow, by feeling for the value of n through means of a subtraction; and so on; but we shall not enumerate further. It is desirable in all calculations, so to arrange the processes, that the *offices* performed by the Variables may be as uniform and fixed as possible.

Supposing that it was desired not only to tabulate B_1, B_3, etc., but A_0, A_1, etc.; we have only then to appoint another series of Variables, V_{41}, V_{42}, etc., for receiving these latter results as they are successively produced upon V_{11}. Or again, we may, instead of this, or in addition to this second series of results, wish to tabulate the value of each successive *total* term of the series (8), viz.: A_0, A_1B_1, A_3B_3, etc. We have then merely to multiply each B with each corresponding A, as produced; and to place these successive products on Result-columns appointed for the purpose.

The formula (8) is interesting in another point of view. It is one particular case of the general Integral of the following Equation of Mixed Differences—

$$\frac{d^2}{dx^2}(z_{n+1}x^{2n+2}) = (2n + 1)(2n + 2)z^n x^{2n}*$$

for certain special suppositions respecting z, x and n.

The *general* integral itself is of the form,

$$z_n = f(n) \,.\, x + f_1(n) + f_2(n) \,.\, x^{-1} + f_3(n) \,.\, x^{-3} + \ldots$$

and it is worthy of remark, that the engine might (in a manner more or less similar to the preceding) calculate the value of this formula upon most *other* hypotheses for the functions in the integral, with as much, or (in many cases) with more, ease than it can formula (8). A. A. L.

* [sic].

APPENDIX 2

EXTRACTS FROM FOUR LETTERS of the Lovelace Papers, published by permission of Lady Wentworth.

1. Babbage to the Countess of Lovelace. 29th December, 1830.

> I have just arrived at an improvement which will throw back all my drawings full six months unless I succeed in carrying out some new views which may shorten the labour.
>
> I have now commenced the description of the engine so that I am fully occupied.

2. Babbage to the Countess of Lovelace. 2nd July, 1843.

> There is still one trifling misapprehension about the variable cards. A variable card may order any number of variables to receive the *same number* upon them at the *same* instant of time. But a variable card never can be directed to order more than *one* variable to be given off at once because the mill could not receive it and the mechanism would not permit it.

3. Babbage to the Countess of Lovelace. 30th June, 1843.

> Only three kinds of variable cards are used.
>
> 1st. Those which give off a variable from the store to the mill and leave zeros for the variable itself.
>
> 2nd. Those which give off a variable from the store to the mill and at the same time (or in the same turn of the hand) retain the same variable in the same place.
>
> 3rd. Those which order any variable in which only zeros exist to receive a result from the mill.

4. Mary Somerville to Lady King. 3rd July, 1836.

> Mr. Babbage is looking wretchedly and has been very unwell. I have done all I could to persuade him to leave town, but in vain. I do fear the machine will be the death of him, for certain I am that the human machine cannot stand that restless energy of mind.

GLOSSARY

Absolute Stop. A method of stopping a machine, see p. 693.

A.C.E. Automatic Computing Engine. The name given to the computer at the National Physical Laboratory.

Aero-elastic effects. Those effects involving interactions between the aerodynamic loading and the structural deformation of an aircraft.

A.M.O.S. A Ferranti Machine which has been bought by the Ministry of Supply.

A.P.E.H.C. All-purpose Electronic Computer built by Birkbeck College Computation Laboratory for the British Tabulating Machine Company.

A.P.E.R.C. All-purpose Electronic Computer built by Birkbeck College Computation Laboratory for the British Rayon Research Association.

A.P.E.X.C. All-purpose Electronic Computer owned and built by Birkbeck College Computation Laboratory.

A.R.C. Automatic Relay Calculator built by Birkbeck College Computation Laboratory for the British Rubber Producers' Research Association.

Bit. A single binary digit, a quantum of information.

Coding. The translation of a micro-programme (q.v.) into detailed instructions specific to the machine being used.

Collator. A device for performing the logical operation *And* on two binary numbers; a machine which sorts two decks of punch cards into one.

Computor. "Bad spelling of *Computer*"—Oxford English Dictionary.

C.P.C. The Card-programmed Electronic Calculator. A computer consisting of a series of standard units which has been made by I.B.M. Large numbers of C.P.C. calculators have been in use for some years, particularly in America. Its limitations were discussed by Lady Lovelace in 1842—more than a hundred years before it was invented (see page 395).

C.R.T. Cathode Ray Tube.

Cybernetics. A word invented by Professor Wiener to describe the field of control and communication theory, whether in the machine or in the animal. None of the authors quite understands what the word means, so it has not been used in this book.

Dekatron. A cold-cathode gas-discharge tube, with ten cathodes and a common anode.

Deuce. The second A.C.E. at the N.P.L.

Division. Many computers do not use a built in divider. Division may be effected by means of calculating and multiplying by the reciprocal of the divisor. The reciprocal may be calculated by *iteration* (see this Glossary) thus. The divisor is first multiplied by a power of two so as to bring it between $\frac{1}{2}$ and 1, and the following iteration performed:

$$x_0 = 1$$
$$x_{n+1} = x_n(2 - kx_n)$$

x_n is an approximation to $1/k$ correct to at least 2^n binary digits.

E.D.S.A.C. Electronic Delay Storage Automatic Computer. The name of the computer at Cambridge University.

E.N.I.A.C. Electronic Numerical Integrator and Calculator. The name given to the computer built by the Moore School of Electrical Engineering in the University of Pennsylvania. The first all-electronic computer to be built.

F.E.R.U.T. The Ferranti Machine now in the University of Toronto.

Flip-flop. A device for "remembering" a single binary digit for an indefinite period (see pages 46–50).

Gestalt. Ehrenfels called attention to man's ability to appreciate certain phenomena which are related to *sets* of stimuli, for example, such qualities as "slenderness," "regularity," "roundness," "angularity." or the characteristic appearance of a circle, a triangle or other geometric shapes. In German the word *Gestalt* is often used as a synonym for form or shape, and Ehrenfels used the term *Gestaltqualitäten* for all of these qualities. Animals with much simpler brain structures than man have this sense of *Gestalt;* a rat may be trained to recognize a circle or a triangle. Dogs have been given nervous breakdowns by training them to distinguish between circles and ellipses and then gradually making the ellipses more and more nearly circular.

Harvard Mk. I. The first modern computer to be built—still working in Harvard. Mark II and III were bought by the U.S. Government. Mark IV is under construction in Harvard.

Hartree Constant. The time which is expected to elapse before a particular electronic computing machine is finished and working. It was Professor Hartree who first pointed out that this estimated time usually remains constant at about six months for a period of several years during the development of a machine. This phenomenon was well known to Babbage. Few engineers are worried unless the "constant" shows signs of increasing monotonically as the years go by.

I.B.M. International Business Machines Corporation.

Iteration. A procedure for calculating a quantity or quantities by successive approximation. The results of each step are used as data for the next step, the calculation being continued until the results of two successive steps are sufficiently nearly equal to be taken as final. The most efficient iterative processes (see, e.g., *Division* in this Glossary) converge with such rapidity as to double the number of correct significant digits at each step.

L.E.O. Lyons Electronic Office. A machine whose design has been based on that of the E.D.S.A.C., and which has been built by Lyons in Cadby Hall to do their commercial calculations.

Loop stop. A method of stopping a machine by making it perform a closed cycle. See Stop (loop).

L.O.R.P.G.A.C. Long Range Proving Ground Automatic Computer.

Macro-programming. The resolution of a programme into separate large-scale processes.

M.A.D.M. Manchester Automatic Digital Machine. The name given to the computer at the Royal Society Computing Laboratory, Manchester University.

Maniac. The name which has been given unofficially to the high-speed machine which is now being built in the Institute for Advanced Studies in Princeton. Alternatively, anyone who has been making or using a digital computer for more than a few years.

Micro-programming. The elaboration of the sections of a macro-programme (q.v.).

Mill. Babbage's name for the arithmetic unit of his machine.

M.O.S.A.I.C. The Ministry of Supply Automatic Integrator and Computer. A machine built by the Post Office for the Ministry of Supply. It's design is based on that of the A.C.E., but it is much bigger; it contains 6,000 valves, and about a quarter of a ton of mercury.

Nimrod. The special-purpose computer for playing the game of nim, exhibited by Ferranti Ltd. at the Festival of Britain Science Exhibition.

N.P.L. The National Physical Laboratory at Teddington.

Optimum programming. An arrangement such that each instruction is extracted during or immediately after the arithmetic operation which precedes it, and numbers needed in the computation are available as soon as they are required.

Parameter. A dimension or other quantity which we can vary for the purpose of the problem.

Poppa. The name popularly given to the I.B.M. selective sequence electronic calculator (S.S.E.C.).

Programme. A list of instructions which causes a digital computer to carry out a series of operations, and solve some particular problem.

Programmer. One who prepares programmes for a machine, "a harmless drudge."

R.A.E. The Royal Aircraft Establishment at Farnborough.

S.E.A.C. The National Bureau of Standards Eastern Automatic Computer, in Washington.

S.S.E.C. The I.B.M. Selective Sequence Electronic Calculator in Madison Avenue, New York.

S.E.C. The Simple Electronic Computer, built by Birkbeck College Computation Laboratory.

Staticisor. A device for converting a number from serial to parallel form, and storing it in parallel form. The alternative spelling "staticizor" is to be deprecated.

Stop, (*loop*). See Loop stop.

Subroutine. A series of instructions for performing a standard process (such as working out sines of angles) which may be used in many programmes.

S.W.A.C. The National Bureau of Standards Western Automatic Computer, in Los Angeles.

T.R.E. Telecommunications Research Establishment, at Malvern.

Türing Machine. In 1936 Dr. Turing wrote a paper on the design and the limitations of computing machines. For this reason they are sometimes known by his name. The umlaut is an unearned and undesirable addition, due, presumably, to an impression that anything so incomprehensible must be Teutonic.

Vector. For our purposes, any set of numbers with a fixed ordering may be considered as a vector.

Whirlwind. The name given to the high-speed computer which is now being built in Massachusetts Institute of Technology.

INDEX

INFORMATION THEORY

And its Engineering Applications

By

D. A. BELL
M.A., B.Sc. (Oxon.), F.I.E.E.
Professor of Electronic Engineering, University of Hull

Fourth Edition

Information theory is the science underlying all exchange of information, and is of especial importance in the development of telecommunications and automatic control systems. The treatment adopted by the author has enabled him to give a clear account of recent developments without requiring a knowledge of mathematics in excess of that which may reasonably be expected from the professional engineer or physicist. His straightforward presentation of the results which have been achieved will save the reader much of the labour involved in studying the original papers, many of which require a knowledge of advanced mathematics for their full understanding. The discussion of the applications of information theory will be found most stimulating, and the author points out that as the new knowledge of this subject becomes more widely disseminated the scope of these applications will continue to grow. 35s. net.

PITMAN